



Sur les erreurs en asservissement visuel

Bernard Espiau

► To cite this version:

| Bernard Espiau. Sur les erreurs en asservissement visuel. RR-2619, INRIA. 1995. inria-00074066

HAL Id: inria-00074066

<https://inria.hal.science/inria-00074066>

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

SUR LES ERREURS EN ASSERVISSEMENT VISUEL

Bernard ESPIAU

N° 2619

Juillet 1995

PROGRAMME 4

 ***apport
de recherche***



SUR LES ERREURS EN ASSERVISSEMENT VISUEL

Bernard ESPIAU

Programme 4 — Robotique, image et vision
Projet BIP

Rapport de recherche n° 2619 — Juillet 1995 — 58 pages

Résumé : En asservissement visuel, la commande appliquée à une caméra utilise un modèle de la dynamique du système, essentiellement représentée par une matrice jacobienne appelée matrice d'interaction. Le calcul de celle-ci fait intervenir les paramètres intrinsèques et extrinsèques de la caméra ainsi que certaines informations sur la géométrie de la scène. L'objet de ce rapport est d'étudier l'influence sur le comportement du système bouclé d'erreurs sur ces paramètres. Cette analyse est effectuée à partir d'une forme générale de la matrice d'interaction, mais reste limitée sur le plan théorique. La majeure partie des résultats est obtenue à l'aide d'études systématiques en simulation et en expérimentation réelle. Par ailleurs, divers points connexes sont abordés, tels les problèmes de détermination de consigne et l'utilisation en stereovision

Mots-clé : Vision par ordinateur, commande de robots, asservissement visuel

(Abstract: pto)

On Errors in Visual Servoing

Abstract: In the classical framework of visual servoing, the control law to be applied to a mobile camera uses a model of the dynamics of the system, which is mainly characterized here by a specific jacobian, the interaction matrix. The computation of this matrix requires knowledge of intrinsic and extrinsic parameters and also some information about 3D in the scene. This report is aimed to study the influence of specific kinds of errors or uncertainties about these parameters on the behavior of the closed-loop system. Although the analysis is performed from a general form of the interaction matrix, the theoretical issues still remain limited. The major part of the results is obtained from systematic experimental studies or simulations. Besides, some other points are addressed, like the determination of set points or the extension to stereovision

Key-words: Computer vision, robot control, visual servoing

1 Introduction

Le problème de la calibration (étalonnage) des caméras en vision artificielle est fréquemment présenté comme à la fois incontournable dès que l'obtention d'informations métriques est demandé, et délicat de par la relative lourdeur des procédures à mettre en oeuvre. Les algorithmes eux-mêmes doivent faire l'objet d'un soin tout particulier si l'on désire obtenir une bonne précision (voir un exemple en [18]). Pour s'affranchir de tels problèmes, plusieurs auteurs se sont récemment intéressés à l'étude de ce qu'il était possible de connaître ou de reconstruire en l'absence de toute calibration par la simple donnée de mise en correspondance d'un certain nombre de points dans un ensemble d'images, en particulier un couple stéréoscopique ([4, 9]). La calibration est alors qualifiée de "faible" et, par exemple, la reconstruction est effectuée au choix d'une base près, ce qui peut être suffisant pour certaines applications.

Dans le cas de l'asservissement visuel, le problème est un peu différent: un vecteur de mesures extraites des images d'un ensemble de caméras (ou plus souvent d'une seule) est utilisé pour calculer à chaque instant une commande, aussitôt appliquée au système, dans le but d'atteindre ou de conserver une situation donnée dans les images. Le passage à la reconstruction 3D n'est pas forcément nécessaire, mais il est par contre important que la boucle ainsi construite présente un comportement acceptable en performance et en fiabilité. Le problème se pose alors en termes de *robustesse* par rapport à des paramètres incertains, c'est à dire à des incertitudes sur la calibration, sur certains aspects liés au 3D ou même sur les consignes dans l'image.

L'objet de ce rapport¹ est d'étudier, principalement expérimentalement, l'influence de telles incertitudes sur le comportement d'un asservissement visuel, en considérant ici uniquement comme primitives géométriques un ensemble de points. En premier lieu, on rappellera les principes de l'approche utilisée. On présentera ensuite une forme générale pour la matrice jacobienne du problème, connue sous le nom de matrice d'interaction, dans laquelle apparaissent explicitement les contributions des diverses sources d'erreurs. On traitera ensuite quelques exemples, et l'on commentera un ensemble de résultats expérimentaux et de simulation. On s'intéressera enfin au problème de la détermination des images cibles en cas de calibration erronée ou absente. En parallèle on évoquera le problème de la calibration faible et de l'utilisation de techniques de reconstruction. Des annexes présentent l'ensemble des résultats ainsi que quelques points techniques.

2 L'approche "asservissement visuel"

Nous ne décrivons pas ici en profondeur cette approche, désormais classique, et renvoyons pour tous détails le lecteur à [2], et [3]. Rappelons en, et de façon simplifiée, les principes nécessaires et suffisants à la compréhension de la suite.

¹Ce travail a été effectué dans le cadre du projet inter-PRC VIA.

2.1 Modélisation

Soit un ensemble de caméras liées rigidement et rendues mobiles par un système quelconque dont la configuration est paramétrée par ψ , $\dim(\psi) = 6$. On suppose que ψ constitue une carte locale de $SE(3)$ et que l'on se situe dans un domaine \mathcal{L} tel que $\forall \bar{r} = f(\psi) \in \mathcal{L} \subset SE(3)$, $\frac{\partial \bar{r}}{\partial \psi}$ soit régulier. Dans le cas où les caméras sont portées par un robot, cela implique que l'on se situe hors de ses singularités géométriques.

On observe dans les images un certain nombre de grandeurs à partir desquelles on construit des "signaux capteurs", $s = \{s_i, i = 1..n\}$. On suppose que ceux-ci ne dépendent que des déplacements relatifs rigides entre les objets de la scène qui les engendrent et les caméras concernées, donc que s peut être modélisée par une application de $SE(3)$ dans R . Notons qu'une extension récente à cette approche permet de prendre en compte des informations liées au mouvement différentiel apparent dans les images (voir [21]), mais nous ne la considérerons pas ici.

Soit maintenant une fonction scalaire $h(s)$ que l'on souhaite minimiser par rapport à ψ . Si un tel minimum m existe, au moins localement, son gradient y sera nul:

$$\forall \psi \in m : \frac{\partial h}{\partial s} \frac{\partial s}{\partial \bar{r}} \frac{\partial \bar{r}}{\partial \psi} = 0 \quad (1)$$

Examinons ces différents termes. Le jacobien du robot $\frac{\partial \bar{r}}{\partial \psi}$ étant supposé régulier, (1) devient:

$$\forall \bar{r} \in f(m) : \frac{\partial h}{\partial s} \frac{\partial s}{\partial \bar{r}} = 0 \quad (2)$$

$\frac{\partial s}{\partial \bar{r}}$, de dimension $n \times 6$ et de rang $p \leq \min(n, 6)$, est notée L^T . La solution de (2) est donc m tel que:

$$\forall \bar{r} \in f(m) : \frac{\partial h}{\partial s} \in \text{Ker}(L^T) \quad (3)$$

Soit par exemple $h = \frac{1}{2}(s - s^*)^T(s - s^*)$. On a alors $\frac{\partial h}{\partial s} = s - s^*$. Suivant la dimension n et le rang p il peut ou non exister plusieurs classes de solution à (3), dont:

$$s - s^* = 0, \quad (4)$$

seule situation que nous considérerons dans la suite.

Si les contraintes définies par (4) sont compatibles, alors (4) définit une *liaison virtuelle* (cf [2]). Lorsque $n > 6$ et si les objets visés sont liés rigidement, cette compatibilité n'est assurée qu'avec un bon choix de s^* . En dérivant (4), il vient:

$$\dot{s} = \frac{\partial s}{\partial \bar{r}} \dot{\bar{r}} = 0 \quad (5)$$

De par la définition de s_i , il apparaît que $\frac{\partial s_i}{\partial \bar{r}}$ peut être assimilé à un élément de se_3^* , dual de l'espace tangent se_3 à $SE(3)$, appelé *torseur d'interaction* H_i . $\dot{\bar{r}}$ est quant à lui le torseur cinématique $\tau = \{V, \Omega\}$. L'équation (5) signifie donc que, comme dans une liaison cinématique réelle, les mouvements localement laissés libres par la contrainte (4) sont dans

le sous-espace réciproque du sous-espace engendré par l'ensemble des torseurs d'interaction $\{H_1, \dots, H_n\}$. La forme matricielle associée à cet ensemble n'est autre que $\frac{\partial s}{\partial \bar{r}} = L^T$, appelée "matrice d'interaction".

2.2 Une approche du contrôle

2.2.1 Préliminaires

Il faut à présent construire une commande qui permette de réaliser la contrainte (4). Considérons pour les besoins de l'exposé un cas très simplifié où la variable de "commande" n'est autre que le torseur cinématique. Supposons s de dimension 6 et les contraintes compatibles et indépendantes. En d'autres termes, dans un certain domaine de $SE(3)$, la solution de (4) est unique et la matrice L^T est régulière. En posant $e = s - s^*$, il est clair que, pour assurer la convergence exponentielle de chaque composante de e vers zero, il faut choisir

$$\tau = -gL^{-T}e, \quad (6)$$

g scalaire positif.

En effet, on a alors $\dot{e} = \frac{\partial s}{\partial \bar{r}}\tau = -ge$. Toutefois, et c'est là le problème qui nous intéresse ici, comme la matrice L^T caractérise l'interaction capteur/environnement, elle dépend des paramètres caméra et de caractéristiques 3D de l'environnement. Ces deux classes de grandeurs peuvent être partiellement ou mal connues. D'où l'utilisation dans la commande d'un modèle (estimé en ligne ou donné a priori), \hat{L}^T . La commande devient alors:

$$\tau = -g\hat{L}^{-T}e \quad (7)$$

L'évolution de e s'écrit:

$$\dot{e} = -gL^T\hat{L}^{-T}e \quad (8)$$

En écrivant:

$$\frac{d}{dt}\left(\frac{1}{2}\|e\|^2\right) = e^T\dot{e} = -ge^TL^T\hat{L}^{-T}e \quad (9)$$

on voit qu'une condition *suffisante* de décroissance de $\|e\|$ est que la matrice $L^T\hat{L}^{-T}$ soit positive (condition C1). Cela se généralise à des schémas plus complexes, faisant en particulier intervenir la dynamique, comme montré en [11].

2.2.2 Forme générale

Considérons à présent le cas où, n étant quelconque et les contraintes restant compatibles, on a: $\text{rang}(L^T) = p < 6$. Les contraintes (4) ne suffisent alors pas à fournir une solution unique. On montre qu'une façon pertinente de poser ce problème consiste à chercher à réguler à zero une fonction e impliquant la satisfaction de (4), tout en profitant des degrés de liberté restants pour minimiser un coût scalaire secondaire h_s représentant par exemple un suivi de trajectoire. Formellement ([11]), cela s'exprime par la régulation de:

$$e = W^\dagger e_1 + \alpha(I_6 - W^\dagger W)\frac{\partial h_s}{\partial \bar{r}} \quad (10)$$

où e_1 est, dans notre cas, une fonction des variables s explicitée plus loin, α un scalaire positif et W une matrice telle que $R(W^T) = R((\frac{\partial e_1}{\partial \bar{r}})^T)$ (propriété P1).

Dans le cas de l'utilisation de capteurs, la fonction e_1 a donc la forme suivante:

$$e_1 = D(s - s^*) \quad (11)$$

où D est une matrice $p \times n$ dite "de combinaison" à déterminer telle que la matrice DL^T soit de rang plein.

Voyons à présent ce que devient la condition de positivité C1 dans ce schéma. On peut montrer que si (condition C2):

$$DL^T W^T > 0 \quad (12)$$

et si α n'est pas trop grand, alors la matrice jacobienne $\frac{\partial e}{\partial \bar{r}}$ est positive. Si l'on choisit maintenant un schéma de commande du type (7), c'est-à-dire de la forme

$$\tau = -g\left(\frac{\partial e}{\partial \bar{r}}\right)^{-1} e, \quad (13)$$

il apparaît que l'on peut assurer la convergence du système en choisissant simplement

$$\left(\frac{\partial e}{\partial \bar{r}}\right)^{-1} = I_6 \quad (14)$$

On a alors:

$$\frac{d}{dt}\left(\frac{1}{2}\|e\|^2\right) = -ge^T \frac{\partial e}{\partial \bar{r}} e < 0 \quad (15)$$

On aura donc par exemple intérêt à choisir $D = WL$ pour satisfaire la condition C2. La propriété P1 devient alors: " W^T est une matrice dont les vecteurs colonnes forment une base du sous-espace engendré par l'ensemble des torseurs d'interaction $\{H_1, \dots, H_n\}$ ".

Comme précédemment, nous devons à présent prendre en compte le fait que L peut être mal connue. La condition C2 s'écrit alors:

$$W\hat{L}L^T W^T > 0 \quad (16)$$

c'est-à-dire (condition C3):

$$\hat{L}L^T \geq 0 \quad (17)$$

A noter que la propriété P1 telle qu'énoncée ci-dessus peut être souvent satisfaite en pratique car la connaissance qu'elle nécessite sur L est minimale et généralement disponible.

En remarquant que $\hat{L}L^T = \hat{L}L^T(\hat{L}^{-T}\hat{L}^T)$, on constate que les conditions C1 et C3 s'expriment finalement sous des formes analogues.

Remarque Une brève approche du cas de l'asservissement visuel en stéréovision est proposée en annexe A.

2.3 Problème considéré

Revenons au problème des erreurs en asservissement visuel. Nous allons restreindre notre étude au cas où le vecteur s ne comporte que des coordonnées rétinienne de points 3D projetés. La matrice L^T est alors la matrice du flot optique correspondant, appelée aussi *jacobien de l'image*. La plupart des erreurs que nous considérons seront concentrées dans cette matrice: incertitudes explicites sur les paramètres, erreurs d'estimation en ligne, etc...

Par ailleurs, nous nous intéresserons à des schémas de commande de type (2) ou (7) suivant les cas. Dans ce cadre, notre objectif sera la régulation autour d'une valeur s^* constante, supposée telle que l'utilisateur considère son objectif comme atteint dès que $s = s^*$. Cela n'est pas anodin: en effet, il n'est ni toujours facile, ni forcément plus sûr d'exprimer un objectif de cette manière, c'est-à-dire sans recourir explicitement à une spécification 3D. Nous reviendrons plus loin sur cet aspect.

3 Formes de la matrice d'interaction

3.1 Calcul direct

Soit R_0 un repère de référence d'origine O_o (repère de commande, souvent appelé repère de l'effecteur) et R_c le repère associé à la caméra concernée et ayant son centre optique O_c pour origine. Soit $s = (u \ v)^T$ le vecteur des coordonnées rétinienne de la projection d'un point \bar{M} de coordonnées $\bar{X}^T = (X \ Y \ Z)^T$ dans R_0 . En utilisant les coordonnées projectives, on peut écrire:

$$q = PM \quad (18)$$

où $q = \delta \begin{pmatrix} s \\ 1 \end{pmatrix}$ et $M = \begin{pmatrix} \bar{X} \\ 1 \end{pmatrix}$, P étant la matrice de projection perspective:

$$P = \begin{pmatrix} P_1 & t \\ p_3^T & t_3 \end{pmatrix} \quad (19)$$

où $\dim(P_1) = 2 \times 3$, $\dim(t) = 2 \times 1$, $\dim(p_3) = 3 \times 1$ et t_3 est scalaire. P est définie à un facteur d'échelle près, et donc par 11 paramètres.

Les équations (18) et (19) donnent:

$$s = \frac{1}{p_3^T \bar{X} + t_3} (P_1 \bar{X} + t) \quad (20)$$

D'où:

$$\dot{s} = \frac{1}{p_3^T \bar{X} + t_3} (P_1 - s p_3^T) \dot{\bar{X}} \quad (21)$$

En écrivant $\dot{\bar{X}} = \vec{V}(O) + \vec{\Omega} \times O\bar{M}$ sous la forme

$$\dot{\bar{X}} = V + As(\bar{X})\Omega \quad (22)$$

où $As(\bar{X})$ est la matrice antisymétrique associée au produit vectoriel par \bar{X} , il vient finalement:

$$L^T = \frac{1}{p_3^T \bar{X} + t_3} (P_1 - sp_3^T) (I_3 \mid As(\bar{X})) \quad (23)$$

Remarque Nous avons considéré ici une projection perspective; on trouvera par contre en (cf [24]) un exemple de calcul de matrice d'interaction dans un cadre affine.

3.2 Forme "canonique"

L'expression (23) permet le calcul explicite de la matrice d'interaction pour une forme quelconque de P . Elle ne se prête toutefois pas aisément à l'analyse. Nous en allons donc en établir une autre version. On sait en effet que toute matrice de projection perspective peut se factoriser de manière unique sous la forme ([13]):

$$P = AP_c D \quad (24)$$

où:

- A est une matrice de changement de coordonnées rétinienne. Elle est de la forme

$$A = \begin{pmatrix} -fk_u & fk_u \cot\theta & u_0 \\ 0 & -\frac{fk_v}{\sin\theta} & v_0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} A_{11} & \alpha \\ 0 & 0 & 1 \end{pmatrix} \quad (25)$$

dont les paramètres, appelés *intrinsèques*, sont:

- f : distance focale
- k_u et k_v : facteurs d'échelle sur les axes u et v
- u_0 et v_0 : coordonnées de l'intersection de l'axe optique et du plan rétinien
- θ : angle entre les axes du repère image

En posant $\alpha_u = fk_u$ et $\alpha_v = fk_v$, il apparaît que A est définie par 5 paramètres indépendants.

Nous ne prenons pas en compte dans ce modèle la distorsion radiale, bien que celle-ci puisse parfois être non négligeable (cas des courtes focales),

- $P_c = (I_3 \mid 0)$ est la matrice normalisée de projection perspective

– D est une matrice 4×4 caractérisant le changement de repère $R_0 \rightarrow R_c$:

$$D = \begin{pmatrix} R_{c0} & [O_c O_0]_c \\ 0 & 1 \end{pmatrix} \quad (26)$$

Les 6 paramètres associés sont appelés *extrinsèques*.

On a donc:

$$q = AP_c D \begin{pmatrix} \bar{X} \\ 1 \end{pmatrix} \quad (27)$$

qui s'écrit aussi:

$$A^{-1}q = P_c \begin{pmatrix} \bar{X}' \\ 1 \end{pmatrix} \quad (28)$$

avec

$$\bar{M}' = \begin{pmatrix} \bar{X}' \\ 1 \end{pmatrix} = D \begin{pmatrix} \bar{X} \\ 1 \end{pmatrix} = DM \quad (29)$$

coordonnées de \bar{M} dans le repère R_c .

D'après (25) on a :

$$A^{-1} = \begin{pmatrix} A_{11}^{-1} & -A_{11}^{-1}\alpha \\ 0 & 1 \end{pmatrix} \quad (30)$$

On a alors, d'après (28), (29) et la définition de q :

$$\begin{pmatrix} A_{11}^{-1}s - A_{11}^{-1}\alpha \\ 1 \end{pmatrix} = P_c \begin{pmatrix} X'/Z' \\ Y'/Z' \\ 1 \\ 1/Z' \end{pmatrix} \quad (31)$$

En posant

$$s' = \begin{pmatrix} u' \\ v' \end{pmatrix} = A_{11}^{-1}s - A_{11}^{-1}\alpha \quad (32)$$

(31) et (32) conduisent à:

$$\dot{s}' = A_{11}^{-1}\dot{s} = \frac{d}{dt} \begin{pmatrix} X'/Z' \\ Y'/Z' \end{pmatrix} \quad (33)$$

où:

$$\frac{d}{dt} \begin{pmatrix} X'/Z' \\ Y'/Z' \end{pmatrix} = L_0^T(\bar{X}') \tau(R_c)|_{R_c} \quad (34)$$

L_0^T étant la forme la plus simple de matrice d'interaction pour un point:

$$L_0^T(\bar{X}') = \begin{pmatrix} -1/Z' & 0 & X'/Z'^2 & X'Y'/Z'^2 & -(1+X'^2/Z'^2) & Y'/Z' \\ 0 & -1/Z' & Y'/Z'^2 & 1+Y'^2/Z'^2 & -X'Y'/Z'^2 & -X'/Z' \end{pmatrix} \quad (35)$$

Les lignes de L_0^T étant des torseurs, il suffit d'appliquer l'opérateur de changement de repère associé pour obtenir la forme générale de matrice d'interaction:

$$L^T = A_{11} L_0^T (\bar{X}') \Theta \quad (36)$$

où \bar{X}' dépend de $D(R_0/R_c)$ et de M à travers (29), et:

$$\Theta = \begin{pmatrix} R_{c0} & -R_{c0} & As[O_c, O_0]_0 \\ 0 & & R_{c0} \end{pmatrix} \quad (37)$$

On peut remarquer que se tromper sur Θ revient à se tromper sur le repère de commande, c'est-à-dire celui auquel on imprime le mouvement τ .

Nous voyons apparaître dans (36) les sources possibles d'incertitudes:

- 6 paramètres extrinsèques dans Θ et L_0^T .
- 3 paramètres intrinsèques dans A_{11} .
- 3 variables 3D pour chaque point M dans L_0 . Si l'on veut se cantonner à l'analyse de la contribution des erreurs de calibration seules, alors on peut supposer, *pour l'étude uniquement*, que ces variables sont connues. Cependant, en pratique, la relation (32) est utilisée dans le calcul de L_0 . D'où l'apparition dans ce cas de la contribution des 5 paramètres intrinsèques dans $L_0^T(Z', s')$.

4 Influence des erreurs

Nous allons étudier à présent successivement les trois types d'erreurs susceptibles d'apparaître dans l'asservissement visuel, en mettant toutefois essentiellement l'accent sur les premières:

1. les erreurs de calibration (paramètres intrinsèques; paramètres extrinsèques);
2. les erreurs dues à une mauvaise détermination de la consigne;
3. les incertitudes sur la géométrie 3D.

Dans chaque cas, on étudiera l'effet de telles erreurs et l'on évoquera si nécessaire des méthodes permettant d'améliorer les résultats.

4.1 Erreurs de calibration

On considère dans cette section le cas d'une seule caméra.

4.1.1 Embryon d'étude analytique

Traitons successivement trois cas très simples.

Exemple 1 Considérons un asservissement sur trois points, dans les conditions suivantes: les paramètres extrinsèques sont connus, ainsi que les coordonnées 3D. Bien que cela soit peu réaliste, supposons que ces dernières sont explicitement utilisées dans le calcul des matrices L_0 . En utilisant la commande (7), avec

$$\hat{L}^T = \hat{A}L_0^T \Theta \quad (38)$$

où

$$\hat{A} = \begin{pmatrix} \hat{A}_{11} & 0 & 0 \\ 0 & \hat{A}_{11} & 0 \\ 0 & 0 & \hat{A}_{11} \end{pmatrix} \quad (39)$$

et

$$L_0^T = \begin{pmatrix} L_0^T(\bar{X}'_1) \\ L_0^T(\bar{X}'_2) \\ L_0^T(\bar{X}'_3) \end{pmatrix} \quad (40)$$

l'équation du système rebouclé est

$$\dot{e} = -gA\hat{A}^{-1}e \quad (41)$$

qui est stable si et seulement si les parties réelles des valeurs propres de la matrice constante $A\hat{A}^{-1}$ sont positives, c'est-à-dire simplement si

- \hat{f} , \hat{k}_u , \hat{k}_v sont de même signe que f , k_u , k_v respectivement; cette condition est évidemment respectée.
- $\frac{\sin \hat{\theta}}{\sin \theta} > 0$. Comme θ est voisin de $\frac{\pi}{2}$, cette dernière condition est toujours vérifiée en pratique.

Exemple 2 Considérons deux points de même 3ème coordonnée $Z^* > 0$ à l'équilibre. Supposons que l'on ne se déplace qu'en translation (mouvement V) et que les seules inconnues soient Z^* et les paramètres intrinsèques. Soient s_1 et s_2 les 4 coordonnées rétinienne des deux projections, avec $s_1 \neq s_2$. Nous supposons de plus que l'on reste proche d'une position d'équilibre (s_1^*, s_2^*) , et que la matrice d'interaction peut donc être considérée comme constante.

En écrivant $e' = \hat{L}(s - s^*)$ et en choisissant la commande $V = -ge'$, g scalaire positif, le comportement résultant du système linéaire $\dot{e}' = -g\hat{L}L^Te'$ est stable si et seulement si les parties réelles des valeurs propres de $\hat{L}L^T$ sont positives.

Comme

$$L^T = \frac{1}{Z} \begin{pmatrix} A_{11} & 0 \\ 0 & A_{11} \end{pmatrix} \begin{pmatrix} -I_2 & A_{11}^{-1}s_1 - u_0 \\ -I_2 & A_{11}^{-1}s_2 - v_0 \end{pmatrix} \quad (42)$$

il vient:

$$\hat{L}L^T = \frac{1}{Z^*\hat{Z}^*} \begin{pmatrix} 2\hat{A}_{11}^T A_{11} & -\hat{A}_{11}^T (s_1^* + s_2^* - (u_0 + v_0)) \\ -(s_1^* + s_2^* - (u_0 + v_0))^T A_{11} & (s_1^* - u_0)^T (s_1 - u_0) + (s_2^* - v_0)^T (s_2^* - v_0) \end{pmatrix} \quad (43)$$

En utilisant le critère de Routh sur le polynôme caractéristique de (43), des conditions sur les paramètres peuvent ainsi être obtenues. Par exemple, si $\theta = \frac{\pi}{2}$, on montre qu'il suffit de choisir des valeurs positives pour les autres paramètres et bien entendu pour \hat{Z}^* pour obtenir une matrice stable à l'équilibre. Le même résultat est obtenu lorsque les points sont symétriques par rapport à l'origine du repère l'image ($s_1^* - u_0 = -s_2^* + v_0$), même lorsque $\theta \neq \frac{\pi}{2}$.

Exemple 3 Soit un ensemble de 3 points non alignés de même 3ème coordonnée $Z^* > 0$ à l'équilibre, conduisant aux mesures s_1, s_2, s_3 dans l'image. On suppose à présent que tous les mouvements sont autorisés, et que la seule inconnue est $Z^* > 0$. Le système étant 6×6 , on peut cette fois étudier la matrice $B = L^T \hat{L}^{-T}$ qui représente la dynamique du système. Comme précédemment, on va considérer la matrice d'interaction comme constante autour de la position d'équilibre. En supposant pour simplifier que la matrice de déplacement D est l'identité et que $u_0 = v_0 = 0$, et après avoir réordonné les lignes de s , B s'écrit à l'équilibre (cf [2]):

$$B = \begin{pmatrix} \lambda I_2 & 0 & 0 \\ 0 & A_{11} \begin{pmatrix} \lambda & 0 \\ 0 & 1 \end{pmatrix} A_{11}^{-1} & 0 \\ 0 & 0 & I_2 \end{pmatrix} \quad (44)$$

où $\lambda = \frac{Z^*}{\hat{Z}^*}$. En utilisant (25) et en posant $\beta = \frac{\alpha_v}{\alpha_u}$, on obtient:

$$A_{11} \begin{pmatrix} \lambda & 0 \\ 0 & 1 \end{pmatrix} A_{11}^{-1} = \begin{pmatrix} \lambda & \frac{\lambda-1}{\beta} \cos \theta \\ 0 & 1 \end{pmatrix} \quad (45)$$

B est donc stable à l'équilibre pour tout $\lambda > 0$. A noter que si l'on s'était intéressé à la positivité de B , on aurait obtenu la condition plus restrictive, mais alors seulement suffisante:

$$0 < \mu^{-2}(1 + \mu^2 - \sqrt{1 + 2\mu^2}) < \lambda < \mu^{-2}(1 + \mu^2 + \sqrt{1 + 2\mu^2}) \quad (46)$$

où $\mu = \frac{\cos \theta}{\beta}$, condition se réduisant à $\lambda > 0$ lorsque $\mu = 0$.

Bien que laissant augurer de bonnes propriétés de robustesse, ces trois exemples demeurent extrêmement simples. En outre, il a été fait dans les deux derniers une approximation (L^T constante). Cependant, nous devons reconnaître humblement qu'il nous a été difficile de pousser un tant soit peu plus loin l'étude analytique. En effet, la condition de positivité générale dans le cas des 3 points s'écrit:

$$AL_0^T \Theta \hat{\Theta}^{-1} \hat{L}_0^{-T} \hat{A}^{-1} > 0 \quad (47)$$

où L_0^T est donnée par (40) et:

$$A = \begin{pmatrix} A_{11} & 0 & 0 \\ 0 & A_{11} & 0 \\ 0 & 0 & A_{11} \end{pmatrix} \quad (48)$$

Les diverses incertitudes interviennent dans (47) d'une manière non triviale. De fait, dès que l'on essaie d'examiner l'influence de l'une d'elles dans le cas général, l'analyse formelle devient vite inextricable. De plus, la condition (47) n'est, rappelons-le, que suffisante, et comme cela est constaté en [2], les domaines de stabilité sont souvent en pratique beaucoup plus grands que ce qu'elle indique. Cela nous a donc incité à approcher l'étude du problème d'une manière expérimentale, telle que décrite dans le paragraphe suivant.

4.1.2 Etude en simulation

Nous avons choisi d'évaluer l'influence des diverses sources d'incertitudes dans un asservissement visuel en utilisant le système de simulation de [2]. Les conditions opératoires sont les suivantes:

Système: On considère quatre points coplanaires formant un carré de côté 20 dans l'espace. Les paramètres de la caméra sont: $A = I_3$; $R_{c0} = I_3$; $[O_c O_0] = 0$. La position de référence est obtenue lorsque les 4 points forment dans l'image un carré centré dont les côtés sont parallèles aux axes, correspondant à une distance $Z^* = 30$

Commande: Elle est de la forme:

$$\tau = -g \hat{L}^{T\dagger} (s - s^*) \quad (49)$$

où $g = 0.1$, $s^{*T} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, -\frac{1}{3}; -\frac{1}{3}, \frac{1}{3}, -\frac{1}{3}, -\frac{1}{3})$, et:

$$\hat{L}^T = \hat{A} \hat{L}_0^T \hat{\Theta} \quad (50)$$

avec:

$$\hat{A} = \begin{pmatrix} \hat{A}_{11} & 0 & 0 & 0 \\ 0 & \hat{A}_{11} & 0 & 0 \\ 0 & 0 & \hat{A}_{11} & 0 \\ 0 & 0 & 0 & \hat{A}_{11} \end{pmatrix} \quad (51)$$

et:

$$\hat{L}_0^T = \begin{pmatrix} \hat{L}_0^T (\hat{Z}'_1, \hat{A}_{11}^{-1} \hat{s}_1) \\ \hat{L}_0^T (\hat{Z}'_2, \hat{A}_{11}^{-1} \hat{s}_2) \\ \hat{L}_0^T (\hat{Z}'_3, \hat{A}_{11}^{-1} \hat{s}_3) \\ \hat{L}_0^T (\hat{Z}'_4, \hat{A}_{11}^{-1} \hat{s}_4) \end{pmatrix} \quad (52)$$

expression dans laquelle on a :

$$\hat{Z}'_i = \hat{a}^T \bar{X}_i + \hat{t}_3 \quad (53)$$

\hat{a}^T étant la dernière ligne de \hat{R}_{c0} et \hat{t}_3 l'estimation de la translation en Z .

Procédure d'évaluation et résultats: On considère deux types de commande: l'une utilise une remise à jour de la matrice d'interaction à chaque pas en fonction des mesures s ; l'autre, généralement moins performante mais plus simple, utilise une valeur constante de \hat{L}^T calculée à l'équilibre s^* (cf [2]). Pour chacune, on évalue individuellement la contribution de chaque erreur paramétrique dans diverses conditions initiales en rotation et en translation. On examine également leurs effets combinés dans quelques cas, soit "extrêmes" soit "raisonnables".

Les résultats sont détaillés en annexe D et l'on peut en tirer les conclusions suivantes:

- l'influence des erreurs d'offset est négligeable;
- une erreur d'un facteur 2 ou même 3 sur la distance focale modifie seulement le temps de convergence dans un rapport voisin, mais ne change pas son allure;
- l'effet d'une erreur sur l'orientation des axes du repère image ne devient perceptible que lorsque celle-ci est bien au-delà d'une valeur réaliste;
- les erreurs sur la situation réelle du repère de la caméra peuvent avoir un effet significatif lorsqu'elles sont suffisamment importantes, par exemple selon les cas 25 à 40 degrés d'erreur de rotation. Concernant la translation, une erreur en z est plus sensible qu'une erreur en x ou y ;
- une erreur sur la rotation introduit une modification globale du comportement transitoire et des couplages nouveaux, alors que, comme déjà mentionné une erreur sur la distance focale joue presque comme un facteur scalaire multiplicatif;
- en cas de grandes erreurs de calibration, la convergence est plus facile à assurer lorsque les écarts initiaux de positionnement sont petits.
- les simulations ne permettent pas de décider si l'un des deux types de commande testés est plus robuste que l'autre: les résultats des séries 6 et 8 sont à cet égard contradictoires.

On résumera donc en indiquant qu'il semble se dégager à la vue de ces résultats une bonne robustesse du système: la convergence est assurée pour des erreurs de calibration raisonnables, voire même assez importantes.

4.1.3 Etude expérimentale

Elle a été menée dans des conditions analogues à celles de l'étude en simulation. Les résultats en sont présentés en annexe C. Ils confirment dans l'ensemble les résultats de simulation, avec toutefois une sensibilité parfois plus grande aux erreurs, même si l'on reste dans des domaines de robustesse très largement acceptables en pratique. On notera en particulier une confirmation de l'effet sur la convergence des erreurs de distance focale. Egalement, le choix entre les deux types de commande reste incertain.

4.2 Erreurs de positionnement final dues aux incertitudes de calibration

Le schéma de contrôle utilisé dans ce rapport a pour objectif l'atteinte d'une valeur s^* . On peut s'interroger sur la façon d'obtenir cette valeur. Une première approche consiste à se placer explicitement et réellement dans la situation correspondant à la satisfaction de l'objectif utilisateur, et à mesurer le vecteur s^* associé. Il s'agit ici en quelque sorte d'étalonner l'application plutôt que la caméra. Une deuxième solution consiste à calculer le s^* à partir du modèle dont on dispose. Mais lorsque l'on va vouloir appliquer une commande au vrai système pour converger vers ce s^* , il n'est d'abord pas certain que cette valeur soit atteignable et, si elle l'est, une erreur de positionnement 3D résultera de l'erreur de calibration. Nous examinons ces deux points dans la suite.

4.2.1 Position du problème

Soit un ensemble de n points 3D de coordonnées euclidiennes \bar{X}_i , les coordonnées rétinienne associées étant s_i . On a l'expression générale habituelle:

$$\delta_i \begin{pmatrix} s_i \\ 1 \end{pmatrix} = AP_c D_c D \begin{pmatrix} \bar{X}_i \\ 1 \end{pmatrix} \quad (54)$$

où D est une matrice de déplacement permettant le changement de repère d'expression des coordonnées des points.

Supposons à présent que les coordonnées rétinienne à atteindre sont calculées, pour les mêmes points 3D de coordonnées \bar{X}_i , à partir d'un modèle de caméra erroné. Nous devons alors chercher s'il existe un ensemble de s_i identiques à ceux de l'équation (54) et un déplacement D' tels que:

$$\delta_i' \begin{pmatrix} s_i \\ 1 \end{pmatrix} = \hat{A}P_c \hat{D}_c D' \begin{pmatrix} \bar{X}_i \\ 1 \end{pmatrix} \quad (55)$$

D'où:

$$AP_c D_c D \begin{pmatrix} \bar{X}_i \\ 1 \end{pmatrix} = \lambda_i \hat{A} P_c \hat{D}_c D' \begin{pmatrix} \bar{X}_i \\ 1 \end{pmatrix} \quad (56)$$

où $\lambda_i = \frac{\delta_i}{\delta_i'}$.

Examinons à présent deux questions.

1- Quand le déplacement dû aux erreurs de calibration est-il indépendant des points utilisés? En posant $D_1 = \{R_1, t_1\} = D_c D$, $D_2 = \{R_2, t_2\} = \hat{D}_c D'$ et $\Phi = \hat{A}^{-1} A$, l'équation (56) s'écrit:

$$\Phi(R_1 X_i + t_1) = \lambda_i (R_2 X_i + t_2) \quad (57)$$

Celle-ci est vérifiée pour tout X_i si et seulement si:

$$\begin{cases} \Phi R_1 = \lambda_i R_2 \\ \Phi t_1 = \lambda_i t_2 \end{cases} \quad (58)$$

La première condition indique que $\lambda_i^{-1} \Phi$ doit être une matrice de rotation. Or, d'après la forme de A (équation (25)), la dernière ligne de Φ est $(0 \ 0 \ 1)$. Nous devons donc avoir $\lambda_i = 1$ pour tout i et Φ matrice de rotation. Cela n'est donc possible que si les seules erreurs de calibration portent sur les paramètres extrinsèques. Alors, $\Phi = I_3$ et les erreurs de positionnement associées sont:

$$\begin{cases} R' = \hat{R}_c^{-1} R_c R \\ t' = \hat{R}_c^{-1} (R_c t + t_c - \hat{t}_c) \end{cases} \quad (59)$$

2- Quel est, pour un ensemble de points donné, le déplacement dû aux erreurs de calibration? En réutilisant les notations précédentes et le partitionnement

$$A = \begin{pmatrix} A_{11} & \alpha \\ 0 & 1 \end{pmatrix}, \quad (60)$$

les équations (54) et (55) donnent respectivement:

$$s_i = A_{11} \frac{R_1 \bar{X}_i + t_1}{r_3^1 \bar{X}_i + t_1^3} + \alpha \quad (61)$$

et

$$s_i = \hat{A}_{11} \frac{R_2 \bar{X}_i + t_2}{r_3^2 \bar{X}_i + t_2^3} + \hat{\alpha} \quad (62)$$

où t_j^i est la i ème composante du vecteur t_j et r_3^j la 3ème ligne de la matrice R_j ($j = 1, 2$).

L'égalité des équations (61) et (62) donne deux équations par point \bar{X}_i , pour 3 inconnues en rotation dans R_2 et 3 en translation dans t_2 . Dans le cas général, il est nécessaire de résoudre ces équations numériquement. Auparavant, il faut s'assurer de l'existence d'une ou plusieurs solutions à ce problème, ce qui n'est pas simple a-priori. Donnons-en une interprétation géométrique.

Partons de l'équation (54) et posons:

$$M_i = D_c D \begin{pmatrix} \bar{X}_i \\ 1 \end{pmatrix} \quad (63)$$

ainsi que, comme en (32)

$$s' = \begin{pmatrix} u' \\ v' \end{pmatrix} = A_{11}^{-1} s - A_{11}^{-1} \alpha \quad (64)$$

L'équation (54) s'écrit alors:

$$\delta_i \begin{pmatrix} s_i' \\ 1 \end{pmatrix} = P_c M_i \quad (65)$$

De façon analogue, (55) peut s'écrire:

$$\delta_i' \begin{pmatrix} s_i'' \\ 1 \end{pmatrix} = P_c D'' M_i \quad (66)$$

où $D'' = \hat{D}_c D' D^{-1} D_c^{-1}$ et:

$$\begin{pmatrix} s_i'' \\ 1 \end{pmatrix} = \Phi \begin{pmatrix} s_i' \\ 1 \end{pmatrix} \quad (67)$$

Il s'agit donc d'étudier le passage de l'équation (65) à l'équation (66), autrement dit l'existence de D'' pour un ensemble de points M_i et une valeur de Φ donnés. Il apparaît que le problème est équivalent à l'énoncé suivant: *étant donné n points 3D et une configuration arbitraire de n points distincts dans l'image, existe-t-il une situation de la caméra telle que les points dans l'image soient les projections des points 3D?* Cette question est illustrée dans les figures équivalentes 1 et 2. Pour $n=2$, il existe une infinité de solutions réelles et pour $n > 4$ en général aucune. Il semble que pour $n = 3$, l'on ne sache encore rien dire de complet sur les conditions d'existence et le nombre des solutions réelles.

Considérons à présent deux cas particuliers:

1- PETITES ERREURS DE CALIBRATION

La matrice Φ est ici supposée proche de l'identité. On l'écrit, sous forme partitionnée:

$$\Phi = \begin{pmatrix} I_2 + E_2 & \phi \\ 0 & 1 \end{pmatrix} \quad (68)$$

où E_2 et ϕ sont supposés petits.

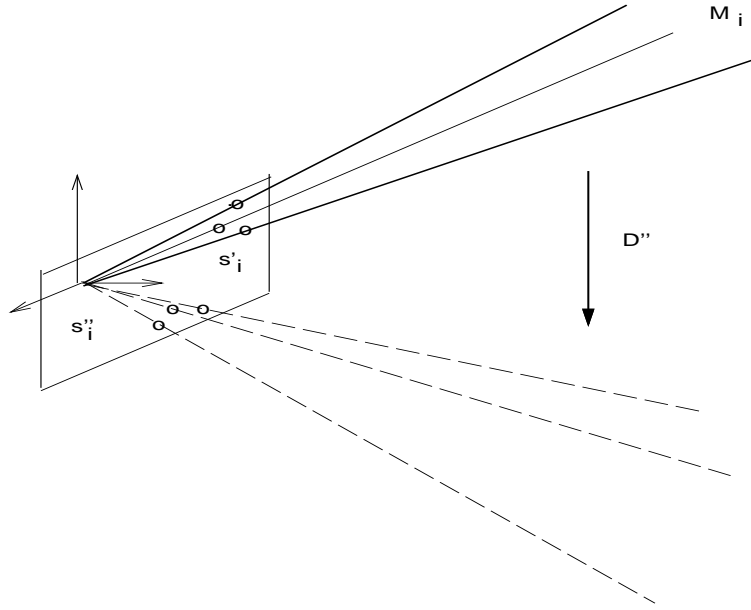


FIG. 1 - : Cas $n = 3$: première interprétation géométrique: mouvement des points 3D

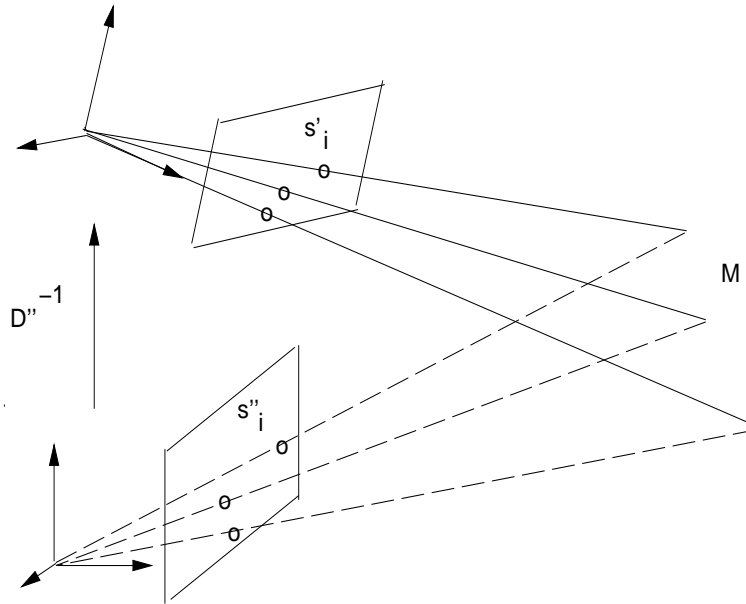


FIG. 2 - : Cas $n = 3$: deuxième interprétation géométrique: mouvement de la caméra

On va chercher le mouvement différentiel qui fait passer de s_i' à s_i'' . On considère uniquement le cas de trois points. Comme la solution existe pour l'ensemble $\{s_1', s_2', s_3'\}$ et que l'on s'intéresse à un petit déplacement, elle existera aussi pour l'ensemble voisin $\{s_1'', s_2'', s_3''\}$ si l'on ne se situe pas initialement sur une singularité telle que caractérisée en [14]. En développant au premier ordre on a :

$$s_i'' = s_i' + T\dot{s}_i' + \dots \quad (69)$$

D'où, en utilisant (68) :

$$\dot{s}_i' \simeq (E_2 s_i' + \phi) \quad (70)$$

Dérivons à présent (65) :

$$\dot{s}_i' = L_0^T(M_i) \begin{pmatrix} V \\ \Omega \end{pmatrix} \quad (71)$$

En posant $\bar{E}_2 = \text{diag}(E_2, E_2, E_2)$, $\epsilon^T = (\phi^T \ \phi^T \ \phi^T)^T$, $s^* = (s_1' \ s_2' \ s_3')^T$ et $L = (L_0(M_1) \ L_0(M_2) \ L_0(M_3))$, il vient finalement :

$$\begin{pmatrix} V \\ \Omega \end{pmatrix} \simeq \frac{1}{T} L^{-T} (\bar{E}_2 s^* + \epsilon) = \frac{1}{T} \begin{pmatrix} b_v \\ b_\Omega \end{pmatrix} \quad (72)$$

Soit comme précédemment $D'' = \{R'', t''\}$ le mouvement cherché. On représente la rotation R'' par son angle θ et son vecteur u : $R'' = \exp(\theta As(u))$. Comme, d'après (72), V et Ω sont à peu près constants, on peut écrire :

$$t'' = \int_0^T V d\tau = b_v \quad (73)$$

et

$$R''(\tau) = \exp(As(\Omega)\tau) \quad (74)$$

D'où : $As(\Omega)T = \theta As(u)$ et, finalement, la rotation cherchée :

$$\begin{cases} \theta = \|b_\Omega\| \\ u = \frac{b_\Omega}{\|b_\Omega\|} \end{cases} \quad (75)$$

2- CAS DE POINTS COPLANAIRES

On sait ([5]) que lorsque plusieurs points 3D appartiennent à un même plan, on a pour chacun la relation, définie à un facteur d'échelle près :

$$\begin{pmatrix} s_i' \\ 1 \end{pmatrix} = H \lambda_i \begin{pmatrix} s_i'' \\ 1 \end{pmatrix} \quad (76)$$

où les s_i'' sont les coordonnées rétinienne obtenues après un déplacement $D = \{R, t\}$ de la caméra, et avec :

$$H = dR + tn^T \quad (77)$$

où n est la normale au plan et d la distance, par rapport au repère avant déplacement. En utilisant (67), l'équation (77) s'écrit :

$$\begin{pmatrix} s_i'' \\ 1 \end{pmatrix} = \lambda_i H \Phi \begin{pmatrix} s_i' \\ 1 \end{pmatrix} \quad (78)$$

et l'on est amené à chercher les solutions de l'équation :

$$\lambda_i H \Phi = I_3 \quad (79)$$

qui s'écrit aussi :

$$R = \frac{1}{d} \left(\frac{1}{\lambda} \Phi^{-1} - tn^T \right) \quad (80)$$

Considérons par exemple le cas où le repère de la caméra est le repère de référence ($D_c D = I_4$) et où le plan est perpendiculaire à l'axe z ($n^T = (0 \ 0 \ 1)$). En raison de la forme de A , la forme générale de Φ^{-1} est :

$$\Phi = \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ 0 & \phi_{22} & \phi_{23} \\ 0 & 0 & 1 \end{pmatrix} \quad (81)$$

En utilisant (81), il apparaît alors que la matrice

$$\frac{1}{\lambda d} \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} - t_1 \lambda \\ 0 & \phi_{22} & \phi_{23} - t_2 \lambda \\ 0 & 0 & 1 - t_3 \lambda \end{pmatrix} \quad (82)$$

doit être une matrice de rotation. On constate alors aisément que la seule solution est l'identité, et qu'elle nécessite d'une part que l'angle θ entre les axes (cf équation (25)) soit connu et égal à $\frac{\pi}{2}$, d'autre part que les coefficients α_u et α_v soient égaux (on note $\alpha_u = \alpha_v = \gamma$). On a alors $\phi_{12} = 0$, $\phi_{22} = \phi_{11} = \frac{\hat{\gamma}}{\gamma}$, $\phi_{13} = \frac{u_0 - u_0}{\gamma}$, $\phi_{23} = \frac{v_0 - v_0}{\gamma}$ et la solution est la translation :

$$\begin{cases} t_1 = d \frac{\phi_{13}}{\phi_{11}} \\ t_2 = d \frac{\phi_{23}}{\phi_{11}} \\ t_3 = d \left(\frac{1}{\phi_{11}} - 1 \right) \end{cases} \quad (83)$$

pour un facteur d'échelle valant $\lambda = \frac{d}{\phi_{11}}$.

4.2.2 En pratique

De l'analyse ci-dessus, il ressort que le calcul de s^* à partir d'un modèle n'est généralement pas la meilleure façon de procéder en présence d'erreurs de calibration. On recommandera donc plutôt en général la détermination *expérimentale* du s^* correspondant à l'objectif exprimé par l'utilisateur.

Il existe cependant une situation où l'on peut résoudre élégamment le problème de la détermination de la consigne dans l'image à partir d'une spécification 3D sans tenir compte de la calibration. Cette méthode a été imaginée par R. Horaud et est utilisée dans [18]. Considérons le cas d'une caméra *non calibrée*, qui observe simultanément deux objets, dont l'un est fixe par rapport à un repère scène et l'autre mobile, c'est-à-dire en fait commandé. Le repère caméra R_c est lié rigidement à l'un *ou* l'autre par un déplacement constant inconnu. Choisissons le cas où la caméra est fixe par rapport à l'objet dans la scène. Cet objet est caractérisé par un ensemble de n points, de coordonnées homogènes M_i connues dans un repère associé, R_S (repère scène).

L'objet mobile (appelons-le *pince*) est caractérisé par m points, de coordonnées homogènes N_j connues dans le repère associé R_p . L'objectif de l'utilisateur (par exemple contrainte d'alignement pour une bonne saisie) est spécifié par la donnée d'une matrice de déplacement désirée, D^* , entre R_p et R_S .

En notant P^p la matrice 4×4 d'une application de l'espace euclidien 3D vers l'espace projectif 3D, définie à un facteur d'échelle près, on peut écrire:

$$M_i^p = \lambda P^p M_i \quad (84)$$

En imposant aux 5 premiers points M_i^p de former la base canonique de l'espace projectif 3D:

$$(0 \ 0 \ 0 \ 1) \ (1 \ 0 \ 0 \ 1) \ (0 \ 1 \ 0 \ 1) \ (0 \ 0 \ 1 \ 1) \ (1 \ 1 \ 1 \ 1) \quad (85)$$

et en ajoutant une contrainte monodimensionnelle sur les coefficients de P^p pour lever l'indétermination, la résolution du système (84) permet d'obtenir P^p . Il est alors possible de calculer l'image des points N_j de la pince dans l'espace projectif 3D par la transformation:

$$N_j^p = \lambda P^p D^* N_j \quad (86)$$

Comme ces opérations sont indépendantes des positions effectives de l'objet et de la pince par rapport à la caméra, elles peuvent être effectuées hors ligne.

L'ensemble $\{M_i^p, i = 1 \dots n; N_j^p, j = 1 \dots m\}$ est appelé *l'invariant projectif* correspondant à l'objectif de positionnement relatif D^* . Le problème est correctement posé si, de plus, $m \geq 4$, pour assurer l'unicité du positionnement relatif, et $n \geq 6$ pour une raison exposée ci-dessous.

Il reste à présent à trouver s^* , c'est-à-dire l'ensemble des coordonnées dans l'image des points de la pince à atteindre étant donnée une image de l'objet visé, de position inconnue a priori. Notons Q la matrice 3×4 de l'application permettant de passer de l'espace projectif 3D à l'espace projectif 2D. Appliquée à l'objet, elle s'écrit:

$$\delta \begin{pmatrix} s_i \\ 1 \end{pmatrix} = Q M_i^p \quad (87)$$

Pour identifier les 12 coefficients de Q , il faut donc connaître 6 points M_i^p et mesurer leurs coordonnées s_i dans l'image. On utilisera les 5 points de la base canonique, et le sixième sera calculé par (84). Une fois Q obtenue, le vecteur objectif est donné par:

$$s_j^* = Q N_j^p \quad j = 1 \dots m \quad (88)$$

Dans le cas où la caméra est fixe dans le repère pince R_p , l'approche précédente reste valide en inversant les rôles des points M et N .

4.3 Effet des incertitudes sur le 3D

Ces incertitudes concernent les coordonnées \bar{X}_i des points de la cible dans le repère R_S (cf. section 3). Sans entrer dans une étude exhaustive de l'état de l'art sur le problème de reconstruction de la géométrie 3D, qui sortirait du cadre d'un rapport focalisé sur les problèmes de calibration, nous pouvons cependant distinguer deux cas principaux.

4.3.1 Calibration connue

Dans cette hypothèse, les incertitudes se ramènent à la méconnaissance des profondeurs Z_i .

Dans le cas de déplacements rigides finis, ce problème peut être résolu en utilisant les techniques de calcul de la géométrie 3D utilisées en stéréovision classique. Il est toutefois nécessaire que la mise en correspondance soit correcte, ce qui est immédiat dans notre cas si nous traitons une image de 4 points. Ainsi, considérons l'un de ces points et notons:

$$\begin{aligned} u_1' &= \frac{X_1'}{Z_1'} \\ v_1' &= \frac{Y_1'}{Z_1'} \end{aligned} \quad (89)$$

où l'indice 1 note l'instant initial, les coordonnées 3D, qui sont requises dans le calcul de L^T , étant définies en (29) et les coordonnées images réduites en (32).

Notons u_2' , v_2' les coordonnées images réduites obtenues après un déplacement rigide supposé connu (mesure du mouvement effectué en mono caméra, ou stéréovision statique calibrée). On note r_k^T ; $k = 1, 2, 3$ les lignes de la matrice de rotation associée et t_1 , t_2 , t_3 les composantes du vecteur de translation. On a alors aisément:

$$Z_1' = \frac{t_1 + t_2 - (u_2' + v_2')t_3}{< (u_2' + v_2')r_3 - r_1 - r_2, (u_1', v_1', 1) >} \quad (90)$$

Cette technique est par exemple utilisée en [8]. Elle fonctionne très bien en stéréovision, mais il faut souligner que lorsqu'elle est appliquée à une séquence *convergente* d'images

monoculaires (et c'est l'objectif visé en asservissement visuel), le système devient mal conditionné, la disparité étant alors très faible.

Lorsque l'on s'intéresse à des mouvements différentiels, il est alors possible d'utiliser des mesures de champs de déplacement pour remonter au 3D. On trouvera un exemple de cette approche en [22].

Cible connue Lorsque les coordonnées des points sont connues dans un repère lié à l'objet ou à la scène, on est ramené à un problème de calcul de "pose", qu'il est même parfois possible d'effectuer à partir d'une seule image. La principale difficulté réside alors dans la mise au point d'algorithmes à la fois simples et robustes. A titre d'exemple, présentons rapidement la méthode proposée dans [18].

Soit les 3 repères:

R_s , lié à l'objet;

R_o , lié à l'effecteur;

R_c , lié à la caméra.

On considère n points de coordonnées \bar{X}_j^s , connues dans R_s . On a par ailleurs, dans R_o :

$$\bar{X}_j^o = D_{os} \bar{X}_j^s \quad (91)$$

et dans R_c :

$$\bar{X}_j^c = D D_{os} \bar{X}_j^s \quad (92)$$

D est la matrice (ici connue) des paramètres extrinsèques, et D_{os} est la transformation à trouver, à partir de la connaissance de \bar{X}_j^s et de la mesure de la projection de \bar{X}_j^c . On notera $\hat{D} = D D_{os}$, dont

$$\hat{D} = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} \quad (93)$$

et

$$\hat{t} = \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix} \quad (94)$$

notent respectivement la matrice de rotation et le vecteur de translation.

Remarque Notons que si l'on applique à chaque instant k à l'effecteur un mouvement relatif connu par rapport à l'objet, $D_v(k)$, on se ramène toujours à chercher un seul déplacement inconnu. On peut en effet écrire, en changeant de repère objet à chaque pas:

$$\bar{X}_j^c(k+1) = D_v^{-1}(k) \bar{X}_j^c(k) \quad (95)$$

qui est donc connu puisque $\bar{X}_j^c(0)$ l'est, et l'équation (92) reste valide.

En utilisant l'équation (32), on calcule à partir des mesures dans l'image les coordonnées dans le plan caméra:

$$\begin{aligned} u'_j &= \frac{X_j^c}{Z_j^c} \\ v'_j &= \frac{Y_j^c}{Z_j^c} \end{aligned} \quad (96)$$

Notons:

$$I = \frac{1}{t_3} r_1 \quad ; \quad J = I = \frac{1}{t_3} r_2 \quad (97)$$

$$u'_0 = \frac{t_1}{t_3} \quad ; \quad v'_0 = \frac{t_2}{t_3} \quad (98)$$

et:

$$\epsilon_j = \frac{1}{t_3} r_3^T \bar{X}_j^s \quad (99)$$

On a alors:

$$\begin{aligned} u'_j &= \frac{I^T \bar{X}_j^s + u'_0}{1 + \epsilon_j} \\ v'_j &= \frac{J^T \bar{X}_j^s + v'_0}{1 + \epsilon_j} \end{aligned} \quad (100)$$

Ces équations se réécrivent:

$$\begin{aligned} (u'_j - u'_0)(1 + \epsilon_j) &= I_p^T \bar{X}_j^s \\ (v'_j - v'_0)(1 + \epsilon_j) &= J_p^T \bar{X}_j^s \end{aligned} \quad (101)$$

où

$$\begin{aligned} I_p &= \frac{1}{t_3} (r_1 - u'_0 r_3) \\ J_p &= \frac{1}{t_3} (r_2 - v'_0 r_3) \end{aligned} \quad (102)$$

et nous avons donc:

$$t_3 = \frac{1}{2} \left(\frac{\sqrt{1 + u'_0{}^2}}{\|I_p\|} + \frac{\sqrt{1 + v'_0{}^2}}{\|J_p\|} \right) \quad (103)$$

A noter que ces équations s'interprètent physiquement par le passage à une approximation paraperspective de la projection.

La résolution proposée dans [18] est itérative: on part de $\epsilon_j(0) = 0$; comme $u'_j - u'_0$ et $v'_j - v'_0$ sont mesurés et \bar{X}_j^s connus, on calcule I_p et J_p par (101), soit en utilisant une pseudo-inverse, soit par une méthode spécifique si les points sont coplanaires; on obtient ensuite t_3 par (103) et t_1 , t_2 par (98); on calcule alors r_3 en utilisant la propriété $r_3 = r_1 \times r_2$ dans (102), puis r_1 , r_2 avec les mêmes équations. On calcule alors $\epsilon_j(1)$ par (99), et l'on itère jusqu'à convergence. Les coordonnées 3D à utiliser dans la matrice d'interaction sont alors obtenues par (92). Dans [18], cette méthode est utilisée avec une caméra fixe qui observe sur la même image la pince d'un robot et l'objet cible. Les paramètres extrinsèques n'interviennent donc pas dans l'analyse.

Une méthode analogue, mais basée sur une reconstruction affine est présentée en [17].

4.3.2 Calibration inconnue

On entre ici dans le domaine de la reconstruction euclidienne à partir des seules images. Celui-ci est l'objet d'une vaste littérature et présente de nombreux problèmes encore ouverts, en particulier parce que la robustesse et la précision des résultats sont parfois peu satisfaisantes.

En ce qui concerne l'asservissement visuel, le principe que l'on peut adopter est simple: il consiste à mettre en oeuvre les techniques de reconstruction émergentes simultanément à l'approche asservissement visuel classique, de façon à améliorer *en parallèle* l'estimation des variables 3D à utiliser dans le calcul de la matrice d'interaction. L'un des avantages majeurs de l'asservissement visuel étant sa simplicité en vue de traitements en temps réel, il importe de ne pas pénaliser cette approche par des calculs trop lourds et non nécessairement justifiés. On privilégiera donc par exemple les techniques récursives, comme le filtre proposé en [16] pour estimer en ligne la matrice essentielle (110) dans le cas où les paramètres intrinsèques sont connus. Les auteurs utilisent cette estimation pour calculer une commande destinée à compenser les mouvements inconnus d'une cible autonome.

5 Conclusion

En automatique, la plupart des commandes utilisent un modèle de la dynamique du processus à contrôler (ici ce modèle est constitué par la matrice d'interaction). Comme cette dynamique n'est pas toujours bien connue, deux approches sont traditionnellement utilisées:

- la première, orientée “robustesse”, cherche à déterminer une loi de commande, utilisant en général un modèle fixé, telle que pour une classe suffisamment large de systèmes ou de perturbations, certaines propriétés de stabilité et, éventuellement de comportement transitoire, soit garanties;
- la deuxième, orientée “autoadaptation”, considère une commande dont la structure (incluant éventuellement un modèle explicite du processus) est fixée, mais dont les paramètres sont estimés en ligne.

Dans ce rapport, nous avons privilégié la première approche. Les résultats obtenus tendent à prouver que, pour les problèmes d'asservissement visuels envisagés ici, point n'est besoin d'une calibration précise, qui n'est d'ailleurs valable que pour une plage de fonctionnement. Des valeurs très approximatives des paramètres intrinsèques sont utilisables, et l'on peut certainement a fortiori se contenter des paramètres fournis par le constructeur. Ce fait est très intéressant, par exemple lorsque l'on utilise des focales variables² ou lorsque les conditions d'environnement sont susceptibles de changer. Ainsi, une modification de l'ouverture du diaphragme qui remet en cause une calibration classique, pourrait être négligée. L'influence des paramètres extrinsèques semble un peu plus importante. La marge de robustesse reste toutefois élevée et largement compatible avec une calibration très grossière. Il

²Une étude d'asservissement visuel en contrôlant explicitement la distance focale est proposée en [23].

reste cependant à déterminer les valeurs de consigne. C'est en fait là que se trouve déplacé le besoin éventuel de précision. Mais on peut remarquer qu'il est souvent plus facile de calibrer une application c'est-à-dire d'amener le système dans la position/orientation 3D souhaitée vis-à-vis de l'environnement et de mémoriser les valeurs correspondantes dans l'image plutôt que d'effectuer une calibration fine de la caméra.

Ces bons résultats d'ensemble incitent donc à penser qu'une calibration approchée, mais "raisonnable", est suffisante dans la plupart des applications. Cela est d'autant plus vrai que l'on se situe dans une perspective de régulation, c'est-à-dire avec de petites erreurs par rapport à la position d'équilibre.

Il existe toutefois des situations dans lesquelles ce point de vue n'est pas satisfaisant. C'est en particulier le cas lorsque les erreurs initiales sont grandes et que le comportement transitoire du système est en lui-même un élément important de l'application. Ceci est en général lié au fait que les trajectoires des points à poursuivre ont une incidence sur les performances du traitement des images au bas niveau: il est en effet nécessaire d'éviter la sortie du champ de vision, de faciliter la mise en correspondance, etc... Des mouvements de trop grande amplitude ou dans de mauvaises directions sont donc à proscrire, et l'on cherchera à se rapprocher le plus possible d'un comportement de type exponentiel découplé. Ceci nécessite de disposer à chaque instant de la meilleure connaissance possible des paramètres de calibration. On mettra donc à profit les mouvements de la caméra pour améliorer les informations disponibles, se situant ainsi dans un contexte "autoadaptatif".

Dans cet ordre d'idée, une première approche possible consisterait, en considérant que, dans le cas de l'asservissement visuel, les mouvements sont petits mais connus, à reconstruire des informations concernant les paramètres de calibration à partir des points dans l'image et de leur champ de déplacement. L'annexe B donne l'expression d'une contrainte portant sur ces grandeurs, qui est l'analogue en mouvement continu de la contrainte épipolaire, et qui pourrait servir de base à un tel algorithme d'estimation.

Une autre possibilité serait de tirer profit des travaux existants en vision par ordinateur, qui cherchent à exploiter uniquement les mises en correspondance de points entre des ensembles d'images obtenues par des déplacements finis pour reconstituer des informations sur la calibration, le mouvement ou le 3D. Celles-ci ne sont pas nécessairement des valeurs numériques explicites de paramètres, mais peuvent être plus globales. Un exemple est celui de la calibration faible. Son principe est le suivant (cf [4],[9],[12]): considérons 2 caméras rigidement liées, les matrices de projection perspective associées P_1 et P_2 étant inconnues. Soit un ensemble d'au moins 8 points pour lesquels il est possible d'établir les mises en correspondance entre les deux images. On considère alors 5 de ces points comme formant la base projective standard dans l'espace 3D et 4 points de leurs projections dans chaque image comme bases projectives dans les espaces rétinien. On montre alors qu'en utilisant seulement les coordonnées rétinien des points, il est possible de déterminer les matrices P_1 et P_2 à une transformation homographique près:

$$\hat{P}_i = P_i G \quad (104)$$

G est la matrice 4×4 de l'application inconnue qui fait passer de la base projective aux coordonnées euclidiennes homogènes des points de l'espace 3D.

On montre par ailleurs qu'il est possible de déterminer la géométrie épipolaire, en particulier d'estimer avec une précision raisonnable la matrice fondamentale F de l'équation (108) (cf annexe A), à partir des seules mesures dans l'image.

Enfin, la reconstruction des points à partir de leurs projections s'avère donc possible sans calibration explicite, au choix du système de coordonnées projectives près.

On a déjà évoqué dans ce rapport l'utilisation de méthodes de ce type, travaillant en espace projectif ou affine, pour améliorer les performances de l'asservissement visuel ([17, 18]). Les premiers résultats sont très prometteurs et cette approche doit faire l'objet dans l'avenir d'investigations plus poussées, de façon à proposer des algorithmes d'asservissement visuel s'affranchissant encore plus des problèmes de calibration.

Remerciements

L'auteur remercie vivement François Chaumette pour avoir réalisé les essais expérimentaux et fourni le logiciel de simulation, et Aristide Santos pour avoir réalisé l'adaptation de ce dernier.

Références

- [1] T. Viéville, O. D. Faugeras *The First Order Expansion of Motion in the Uncalibrated Case*
Computer Vision and Image Understanding, to appear, 1995
- [2] F. Chaumette *La relation vision-commande: théorie et application à des tâches robotiques*
Thèse, Université de Rennes, France, juillet 1990
- [3] B. Espiau, F. Chaumette, P. Rives *A New Approach to Visual Servoing in Robotics*
IEEE Trans. on Robotics and Automation, vol 8 no 3, june 1992
- [4] O. D. Faugeras *What can be seen in three dimensions with an uncalibrated stéréo rig?*
European Conference on Computer Vision, Genova, Italy, 1992
- [5] O. Faugeras, F. Lustman, G. Toscani *Calcul du mouvement et de la structure à partir de points et de droites*
Rapport INRIA no 661, Mai 1987
- [6] J.T. Feddema, O.R. Mitchell *Vision-guided servoing with feature-based trajectory generation*
IEEE Trans. on Robotics and Automation, vol 5 no 5, october 1989
- [7] R. Haralick *Determining camera parameters from the perspective projection of a rectangle*
Pattern Recognition, Vol 22, n.3, pp 225-230, 1989

-
- [8] N. Maru, H. Kase, S. Yamada, A. Nishikawa *Manipulator Control by Visual servoing with the Stereo Vision*
Int. Conf. on Intelligent Robots and Systems, Yokohama, Japan, July 26-30, 1993
 - [9] R. Mohr, L. Quan, F. Veillon, B. Boufama *Relative 3D Reconstruction Using Multiple Uncalibrated Images*
Research Report LIFIA/IMAG RT84, Grenoble, France, 1992
 - [10] P. Rives, F. Chaumette, B. Espiau *Visual Servoing Based on a Task-function Approach*
1st International Symposium on Experimental Robotics, Montreal, Canada, June 1989
 - [11] C. Samson, M. Le Borgne, B. Espiau *Robot Control: the Task Function Approach*
Clarendon Press, Oxford University Press, Oxford, UK, 1990
 - [12] A. Shashua *Projective Structure from two Uncalibrated Images: Structure from Motion and Recognition*
AI Memo no 1363, MIT, Cambridge, USA, Sept 1992
 - [13] Q.T. Luong *Matrice Fondamentale et Calibration Visuelle sur l'Environnement: Vers une plus Grande Autonomie des Systemes Robotiques*
Thèse, Université de Paris-Sud Centre d'Orsay, Decembre 1992
 - [14] H. Michel, P. Rives *Singularities in the determination of the situation of a robot effector from the perspective view of 3 points*
Rapport INRIA no 1850, Février 1993
 - [15] B.H. Yoshimi, P.K. Allen *Active Uncalibrated Visual Servoing*
Proc. IEEE Int. Conf. on Robotics and Automation, pp 156-161, 1994
 - [16] S. Soatto, P. Perona *Structure-independent Visual Motion Control on the Essential Manifold*
Proc. 4th IFAC Symposium on Robot Control, Capri, Septembre 1994
 - [17] N. Andreff *Asservissement Visuel et Calibration Faible*
Rapport de DEA d'Informatique, Grenoble, Juin 1995
 - [18] R. Horaud, F. Dornaika, C. Bard, B. Espiau *Visually Guided Object Grasping*
soumis à IEEE Trans. on Robotics and Automation, special issue on "Vision-based Control of Robot Manipulators"
 - [19] R. I. Hartley *Euclidian Reconstruction from Uncalibrated Views*
Applications of Invariance in Computer Vision, Springer-Verlag, 1994
 - [20] R. Horaud, F. Dornaika *Hand-Eye Calibration*
International Journal of Robotics Research, 14 (3), June 1995

-
- [21] V. Sundareshwaram, P. Bouthemy, F. Chaumette *Visual Servoing using Dynamic Image Parameters*
Rapport de recherche IRISA no 854, Rennes, août 1994
 - [22] P. Rives, B. Espiau *Estimation récursive de primitives 3D au moyen d'une caméra mobile* Traitement du Signal, vol 4 no 3, 1987
 - [23] B. Espiau *Asservissement visuel et contrôle de la distance focale* Rapport de Recherche INRIA no 2613, juillet 1995
 - [24] C. Colombo, B. Allotta, P. Dario *Experimental Validation of an Active Control Scheme Based on a Reduced Set of Image Parameters* 4th Int. Symp. on Experimental Robotics, Stanford, USA, 30/6-2/7 1995 Rapport de Recherche INRIA no 2613, juillet 1995 Rapport de Recherche INRIA no 2613, juillet 1995

ANNEXES

A Asservissement visuel en stéréovision

Cas d'un point Considérons deux caméras liées rigidement, de repères associés R_c^i , et dont les matrices de projection sont $P_i = A_i P_c D_i$, $i = 1, 2$. Le repère de référence est R_0 .

Soit un point 3D $M = \begin{pmatrix} \bar{X} \\ 1 \end{pmatrix}$ dans R_0 . On considère les 4 coordonnées rétiniennees $\tilde{s} = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$ de ses projections. La matrice d'interaction associée, de dimension 4×6 , est:

$$\tilde{L}^T = \begin{pmatrix} A_{11}^1 L_0^T(Z'_1, u'_1, v'_1) \Theta_1 \\ A_{11}^2 L_0^T(Z'_2, u'_2, v'_2) \Theta_2 \end{pmatrix} \quad (105)$$

où

$$\begin{pmatrix} u_i \\ v_i \end{pmatrix} = A_{11}^i \begin{pmatrix} u'_i \\ v'_i \end{pmatrix} + \alpha \quad (106)$$

Les coordonnées Z'_1 et Z'_2 correspondent au même point M . Elles sont donc liées par la relation:

$$Z'_2 = a_{12}^T M'_1 \quad (107)$$

où a_{12}^T est la troisième ligne de la matrice $D_2 D_1^{-1}$, et $M'_i = D_i M$.

Par ailleurs, s_1 et s_2 sont liés par la contrainte épipolaire:

$$c_e(\tilde{s}) = q_1^T F q_2 = 0 \quad (108)$$

où $q_i = \delta_i \begin{pmatrix} s_i \\ 1 \end{pmatrix}$, et F est la *matrice fondamentale*. Cela indique que nous devons contrôler un vecteur de taille 3 et non 4.

Il est connu (voir [13] par exemple) que F , de rang 2, est de la forme:

$$F = A_2^{-T} E A_1^{-1} \quad (109)$$

avec

$$E = A_s [O_c^2, O_c^1]_2 R_{21} \quad (110)$$

où R_{21} est la matrice de rotation de $D_2 D_1^{-1}$. E est appelée la *matrice essentielle*.

En écrivant F sous la forme

$$F = \begin{pmatrix} \Psi & \psi_1 \\ \psi_2^T & \psi_3 \end{pmatrix} \quad (111)$$

où $\dim(\Psi) = 2 \times 2$, et ψ_3 est scalaire, il vient:

$$c_e(\tilde{s}) = s_1^T \Psi s_2 + \psi_2^T s_2 + \psi_1^T s_1 + \psi_3 = 0 \quad (112)$$

Soit

$$h^T = \frac{\partial c_e}{\partial \tilde{s}} = \left(s_2^T \Psi^T + \psi_1^T \mid s_1^T \Psi + \psi_2^T \right) \quad (113)$$

En dérivant (112) il vient:

$$\dot{c}_e(\tilde{s}) = h^T \dot{\tilde{s}} = h^T \tilde{L}^T \tau = 0 \quad (114)$$

Comme (114) doit être vraie pour tout τ , on a alors

$$h^T \tilde{L}^T = 0 \quad (115)$$

ce qui confirme au passage que, sauf si $h = 0$, \tilde{L}^T est bien de rang 3.

On peut donc choisir de contrôler trois variables indépendantes, que l'on prend de la forme

$$\sigma = C \tilde{s} \quad (116)$$

$\dim(C) = 3 \times 4$, C telle que $C \tilde{L}^T$ soit de rang plein. Dérivons (116):

$$\dot{C} \tilde{s} + C \dot{\tilde{s}} = \dot{\sigma} \quad (117)$$

En associant (115) et (117), il vient:

$$\begin{pmatrix} h^T \\ C \end{pmatrix} \dot{\tilde{s}} = \begin{pmatrix} 0 \\ \dot{\sigma} - \dot{C} \tilde{s} \end{pmatrix} \quad (118)$$

Pour éviter de rencontrer une singularité (chute de rang) de $C \tilde{L}^T$, une condition nécessaire est donc:

$$\det \begin{pmatrix} h^T \\ C \end{pmatrix} \neq 0 \quad (119)$$

On choisira par exemple

$$C = \Pi \begin{pmatrix} I_3 & 0 \end{pmatrix} \quad (120)$$

ou Π est une matrice de permutation de colonnes telle que la composante de h correspondant à la colonne nulle de C soit non nulle.

Si l'on ne désire pas explicitement sélectionner explicitement trois composantes de \tilde{s} , il faut utiliser l'approche décrite au paragraphe 2.2.2. La condition de positivité C2 est satisfaite en choisissant pour la matrice D de

$$e_1 = D(\tilde{s} - s^*) \quad (121)$$

l'expression

$$D = W\tilde{L}B \quad (122)$$

où B est une matrice $4 \times 4 \geq 0$, de rang ≥ 3 , telle que D soit de rang 3. Rappelons que W est ici une matrice 3×6 de rang plein telle que $R(W^T) = R(\tilde{L})$. Notons que cette dernière condition est elle-même vérifiée si l'on peut écrire $W = C\tilde{L}^T$, C satisfaisant (119). La condition (119) est alors équivalente à

$$\det \begin{pmatrix} h^T \\ W\tilde{L}B \end{pmatrix} \neq 0 \quad (123)$$

Remarque

Les deux approches sont équivalentes si l'on choisit $B = C^T C$: soit $\sigma = C(\tilde{s} - s^*)$. On cherche à construire la tâche $e_1 = D'\sigma$. Comme $\frac{\partial \sigma}{\partial r} = c\tilde{L}^T$, le choix $D' = W\frac{\partial \sigma}{\partial r}$ conduit à $D' = W\tilde{L}C^T$. D'où:

$$e_1 = D'\sigma = W\tilde{L}C^T C(\tilde{s} - s^*) \quad (124)$$

à comparer à (121),(122).

Cas de plusieurs points Remarquons en premier lieu qu'avec deux points et une paire stéréo, il reste un degré de liberté car l'on doit tenir compte de deux contraintes épipolaires et d'une contrainte de rigidité. La matrice L^T , de dimension 8×6 , est de rang 5. Une démarche analogue à la précédente peut ainsi être utilisée. Ne la détaillons pas et passons au cas où l'on dispose de $p \geq 3$ points, tels que rang $(\tilde{L}) = 6$. La fonction à réguler peut alors être de la forme:

$$e = \tilde{L}(\tilde{s} - \tilde{s}^*) \quad (125)$$

où s comprend $4p$ éléments. Un choix possible, cohérent avec la situation habituelle en monoculaire, est:

$$e = \hat{\tilde{L}}_{|\tilde{s}=\tilde{s}^*}(\tilde{s} - \tilde{s}^*) \quad (126)$$

Avec la commande $\tau = \lambda e$, le système est stable si la condition

$$\hat{\tilde{L}}_{|\tilde{s}=\tilde{s}^*} \tilde{L}^T > 0 \quad (127)$$

est satisfaite.

Notons qu'un autre choix possible, assurant un meilleur découplage est (cf [2]):

$$e = (\hat{\tilde{L}})^\dagger_{|\tilde{s}=\tilde{s}^*}(\tilde{s} - \tilde{s}^*) \quad (128)$$

Remarquons que la condition (127) s'écrit également:

$$\sum_{j=1}^p (\hat{L}_j^1 L_j^{1T}) + \sum_{j=1}^p (\hat{L}_j^2 L_j^{2T}) > 0 \quad (129)$$

où L_j^i est la matrice d'interaction correspondant au point j pour la caméra i . Elle est donc satisfaite si chacun des termes, correspondant chacun à une caméra, est positif.

B Un invariant différentiel

La contrainte épipolaire (108) est un invariant par rapport au 3D en ce sens qu'elle exprime la relation entre 2 projections du même point après un déplacement rigide comme dépendant seulement de ces projections et des paramètres de calibration. Dans ce paragraphe, nous recherchons une expression analogue pour le mouvement continu, en exprimant la relation qui lie les points dans l'image, leur champ de déplacement et les paramètres de calibration indépendamment du 3D. D'où l'expression pompeuse d'"invariant différentiel".

Nous reprenons les notations de la section 3.2. On note, de plus:

$$m = \begin{pmatrix} s \\ 1 \end{pmatrix} \quad m' = \begin{pmatrix} s' \\ 1 \end{pmatrix} \quad (130)$$

où s' est défini en (32).

En utilisant (29), et en posant $U^T = (0 \ 0 \ 1)$, nous pouvons donc écrire

$$Z'm = A\tilde{X}' \quad (131)$$

où $Z' = U^T \tilde{X}'$, et qui se dérive en:

$$Z'\dot{m} = -\dot{Z}'m + A\dot{\tilde{X}}' \quad (132)$$

Comme $\dot{Z}' = U^T \dot{\tilde{X}}'$, nous obtenons:

$$Z'\dot{m} = (A - mU^T)\dot{\tilde{X}}' \quad (133)$$

En utilisant (32), il est immédiat d'obtenir:

$$A - mU^T = A_{11} \begin{pmatrix} I_2 & -s' \\ 0 & 0 \end{pmatrix} \quad (134)$$

En notant $\{V, \Omega\}$ le torseur cinématique décrivant le mouvement du repère caméra, nous pouvons finalement écrire, avec (131,133):

$$Z'\dot{m} = A_{11} \begin{pmatrix} I_2 & -s' \\ 0 & 0 \end{pmatrix} (V + \tilde{\Omega}Z'A^{-1}m) \quad (135)$$

où $\tilde{\Omega}$ note la matrice antisymétrique associée au produit vectoriel $\Omega \times$. En utilisant (30), (32) et (130), on peut donc écrire:

$$\dot{m} = \frac{1}{Z'} A_{11} \begin{pmatrix} I_2 & -s' \\ 0 & 0 \end{pmatrix} V + A_{11} \begin{pmatrix} I_2 & -s' \\ 0 & 0 \end{pmatrix} \tilde{\Omega}m' \quad (136)$$

Ces 3 équations se réduisent en fait à 2, la dernière ligne étant nulle. En éliminant Z' ³ et en utilisant (130), il vient:

³Pour cela, il suffit de remarquer que si $y = ax + b$, avec $\dim(y, x, b) = 2$ et a scalaire non nul, alors on a: $(y - b)^T Gx = 0$, G étant donné en (138).

$$\dot{s}^T G A_{11}(I_2 - s')V + m'^T \tilde{\Omega} \begin{pmatrix} I_2 \\ -\dot{s}'^T \end{pmatrix} A_{11}^T G A_{11}(I_2 - s')V = 0 \quad (137)$$

où l'on a posé

$$G = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad (138)$$

On remarque que dans le deuxième terme de cette expression on peut écrire:

$$\begin{pmatrix} I_2 \\ -\dot{s}'^T \end{pmatrix} A_{11}^T G A_{11}(I_2 - s')V = \det(A_{11})\tilde{m}'V = -\det(A_{11})\tilde{V}m' \quad (139)$$

Par ailleurs, dans le premier terme de (137), on peut écrire en utilisant (32):

$$\dot{s}^T G A_{11} = \dot{s}'^T A_{11}^T G A_{11} = \det(A_{11})\dot{s}'^T G \quad (140)$$

En reportant (140) et (139) dans (137), et en simplifiant par $\det(A_{11})$, il vient finalement:

$$\dot{s}'^T G (I_2 - s')V - m'^T \tilde{\Omega} \tilde{V}m' = 0 \quad (141)$$

que l'on peut écrire également en utilisant la notation de (32):

$$F^T(m', \dot{m}')V + m'^T \tilde{\Omega} \tilde{V}m' = 0 \quad (142)$$

où

$$F = \begin{pmatrix} \dot{v}' \\ -\dot{u}' \\ -u'\dot{v}' + v'\dot{u}' \end{pmatrix} \quad (143)$$

Considérons à présent un changement de repère correspondant aux paramètres extrinsèques et se traduisant par l'application de l'inverse de l'opérateur (37) au torseur $\tau = \{V, \Omega\}$ pour obtenir le torseur de commande $\tau_c = \{V_c, \Omega_c\}$. On notera cet opérateur simplement:

$$\Theta = \begin{pmatrix} R & -E \\ 0 & R \end{pmatrix} \quad (144)$$

où $E = R\tilde{t}$ est la matrice essentielle, et

$$\tau = \Theta\tau_c \quad (145)$$

A partir de (145), en utilisant le fait que, lorsque R est une matrice de rotation, on a $R(A \times B) = (RA) \times (RB)$, et en développant le double produit vectoriel, on peut écrire:

$$\Omega \times V = R(\Omega_c \times V_c) - \|\Omega_c\|^2 R t + \langle \Omega_c, t \rangle R \Omega_c \quad (146)$$

et donc:

$$\tilde{\Omega}\tilde{V} = R\tilde{\Omega}_c\tilde{V}_c - \|\Omega_c\|^2 E + \langle \Omega_c, t \rangle R\tilde{\Omega}_c \quad (147)$$

et l'équation (142) devient finalement:

$$\dot{s}'^T G (I_2 - s')(RV_c - E\Omega_c) + m'^T (R\tilde{\Omega}_c\tilde{V}_c - \|\Omega_c\|^2 E + \langle \Omega_c, t \rangle R\tilde{\Omega}_c)m' = 0 \quad (148)$$

C Résultats expérimentaux

C.1 Conditions Expérimentales

La caméra est montée sur un poignet à trois rotations, lui-même installé sur un portique à trois axes en translation. La cible est constituée de quatre pastilles blanches sur un fond noir, dont les centres forment un carré de 20 cm de côté. La position initiale est telle que l'orientation entre le repère caméra et le repère objet est $(-20, 10, 30)$ degrés et la translation $(5, 5, 50)$ cm. La position finale est telle que le carré apparaisse centré dans l'image et à une distance de 30 cm de la caméra. Les images correspondantes sont données figure 3.

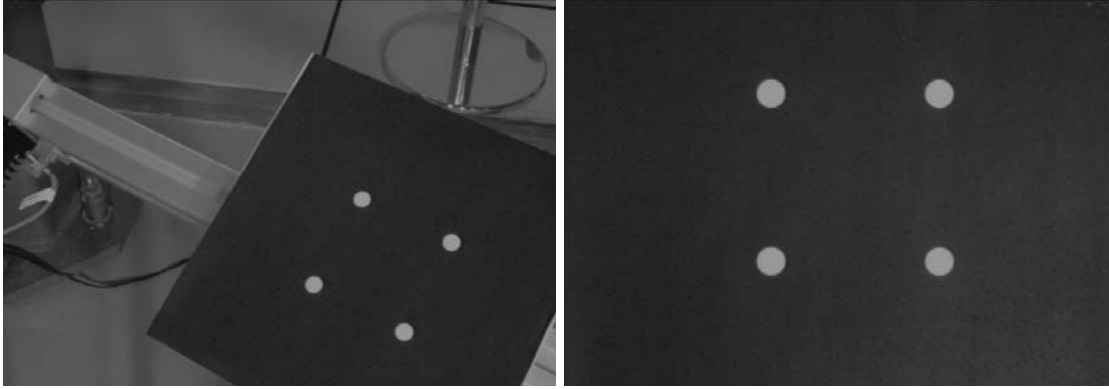


FIG. 3 - : Images initiale et finale (512×730 pixels)

La période d'échantillonnage est de 40 ms et le gain g vaut 0,5. Les paramètres intrinsèques estimés par calibration ont les valeurs suivantes :

$$u_0 = 383, v_0 = 234, -fk_u = 1148.9, -fk_v = 1149.2$$

Deux types de commandes ont été utilisés: la première calcule la matrice d'interaction à chaque pas; en effet, connaissant les dimensions du carré cible, il est possible de calculer à partir de l'image sa position et son orientation par rapport à la caméra, donc les paramètres 3D nécessaires. [7]). La seconde utilise la valeur de la matrice à la position finale désirée.

Les résultats présentés dans la suite sont tous au format suivant:

- à gauche, la valeur mesurée à chaque itération de $s - s^*$ (en pixels) ;
- au milieu, la valeur mesurée à chaque itération de $\|s - s^*\|$;
- à droite, la valeur calculée à chaque itération de τ (en cm/s et dg/s) ;

C.2 Résultats

C.2.1 Courbes de référence

Les figures 4 et 5 donnent respectivement les comportements obtenus avec les deux types de commandes dans le cas où il n'y a pas d'erreur introduite sur les paramètres de calibration. On notera que, dans ce cas, la première conduit à une convergence plus rapide et à un meilleur découplage. Ce ne sera pas toujours vrai par la suite.

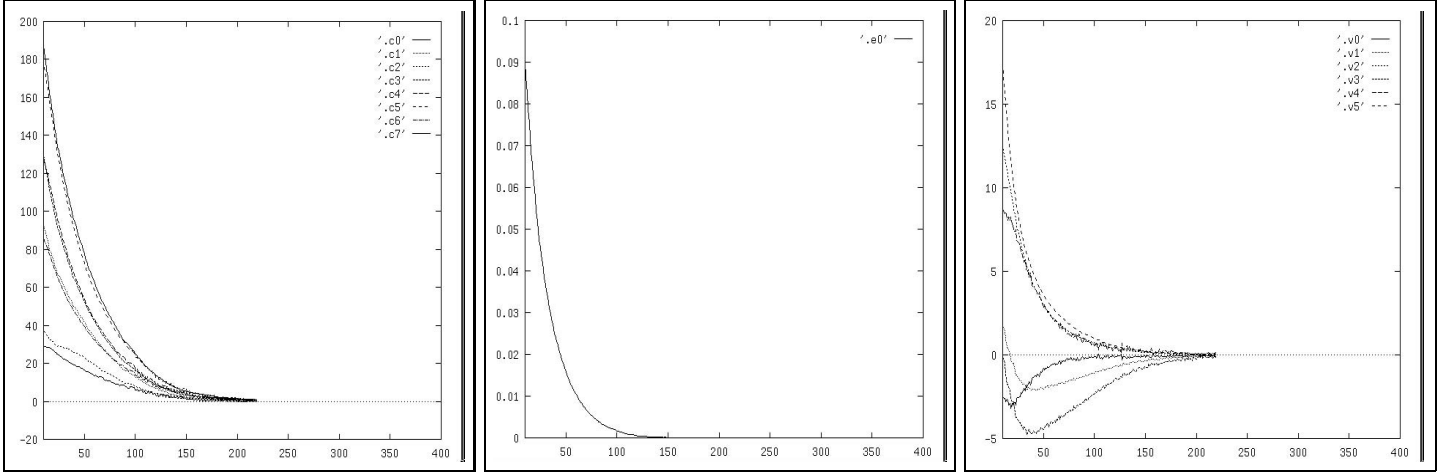


FIG. 4 - : Calcul de L^{-T} à chaque iteration

C.2.2 Erreurs sur la distance focale

Les figures 6 et 7 ont été obtenues en divisant par 2 la distance focale (soit $-fk_u = -fk_v = 575$). Les figures 8 et 9 sont obtenues en augmentant de 50% la distance focale (soit $-fk_u = -fk_v = 1725$).

On note dans tous les cas un effet très net sur la durée de convergence.

C.2.3 Erreurs sur les paramètres extrinsèques

Les figures 10 et 11 (respectivement 12 et 13) ont été obtenues en ajoutant 5 cm à la composante de translation en x (respectivement en z) de la matrice de passage robot-caméra.

Les figures 14 et 15 (respectivement 16 et 17) ont été obtenues en ajoutant une rotation de 5 deg autour de l'axe x (respectivement autour de l'axe z) de la matrice de passage robot-caméra.

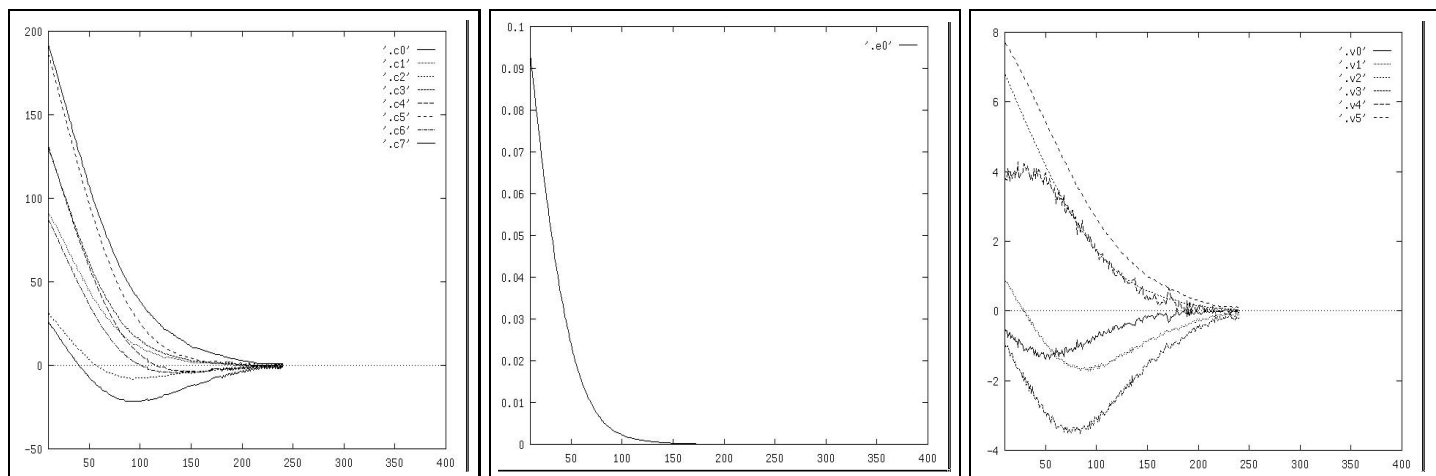


FIG. 5 - : Matrice L^{-T} constante ($= L|_{s=s^*}^{-T}$)

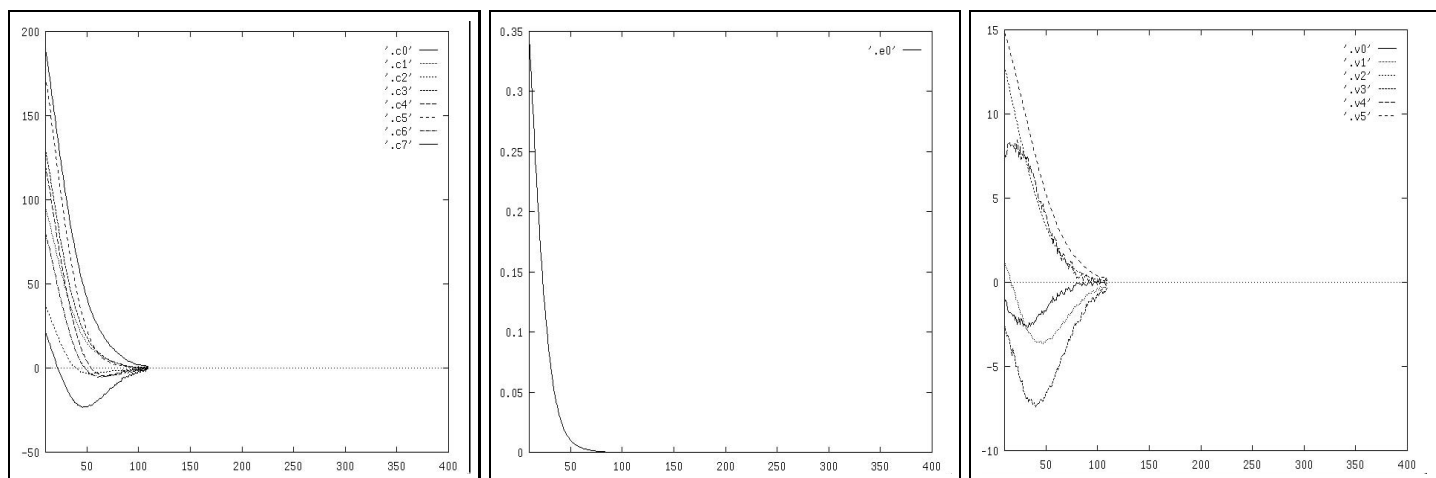


FIG. 6 - : Matrice constante, erreur sur la focale (-50%)

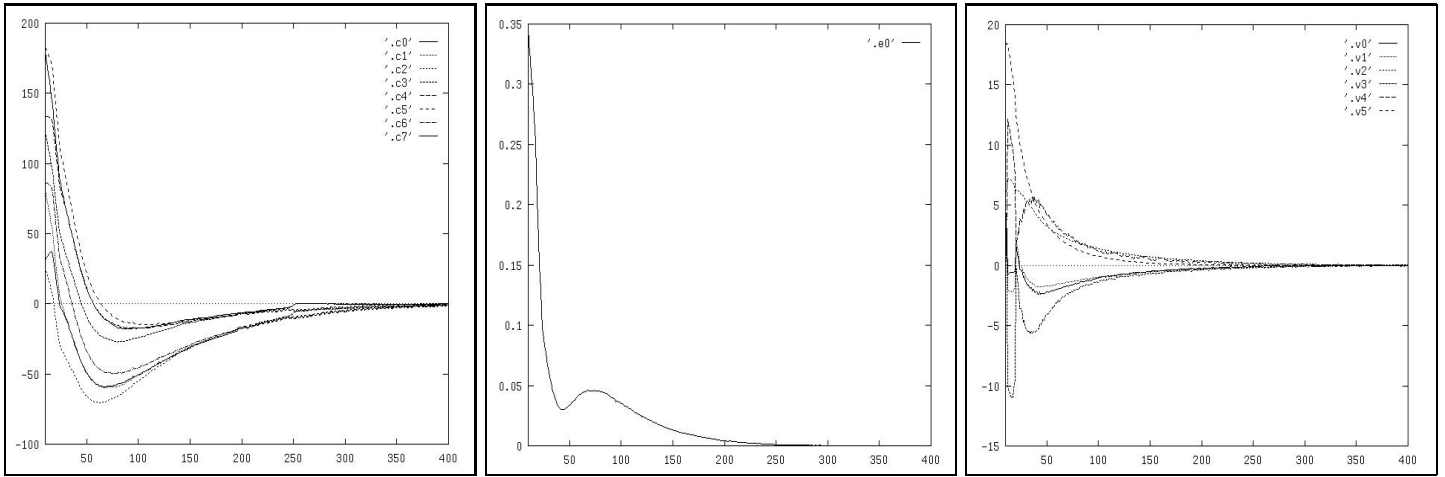


FIG. 7 - : Matrice non constante, erreur sur la focale (-50%)

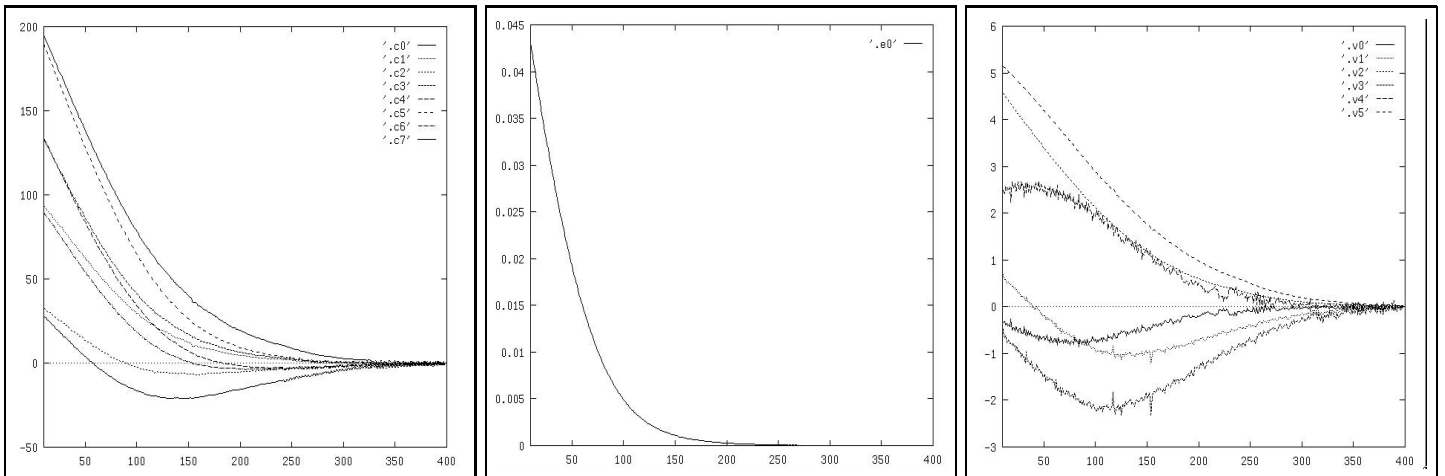


FIG. 8 - : Matrice constante, erreur sur la focale (+50%)

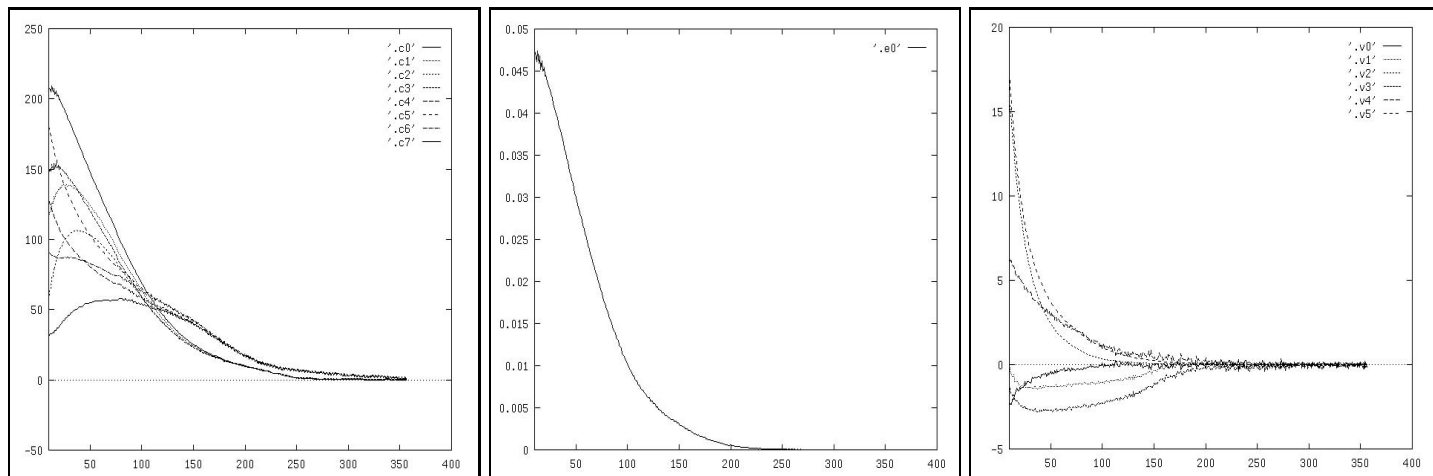
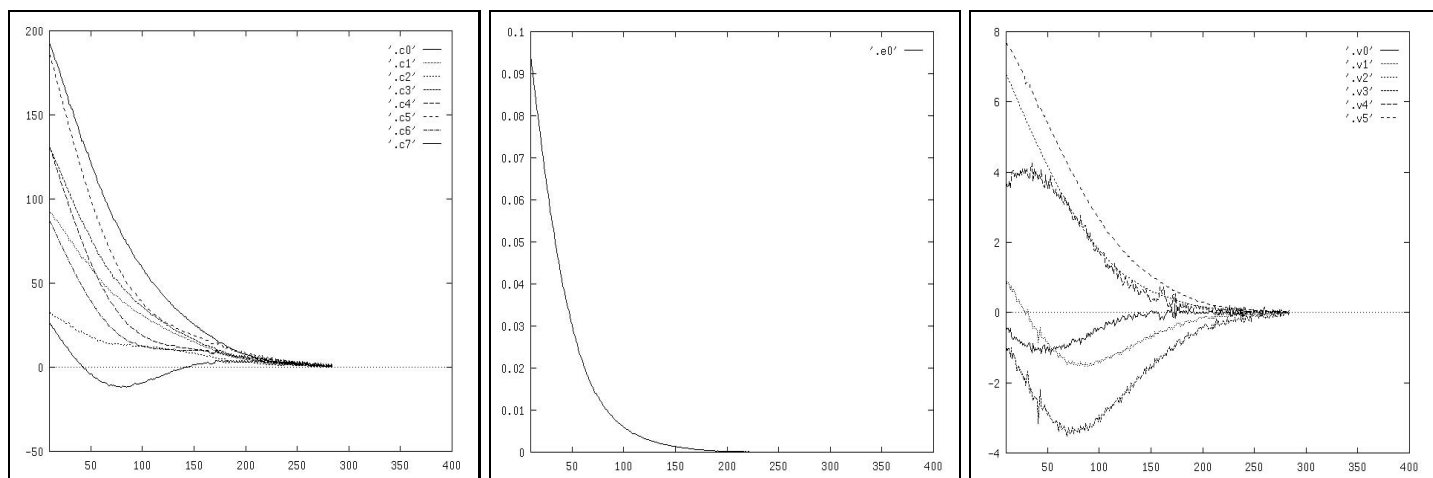
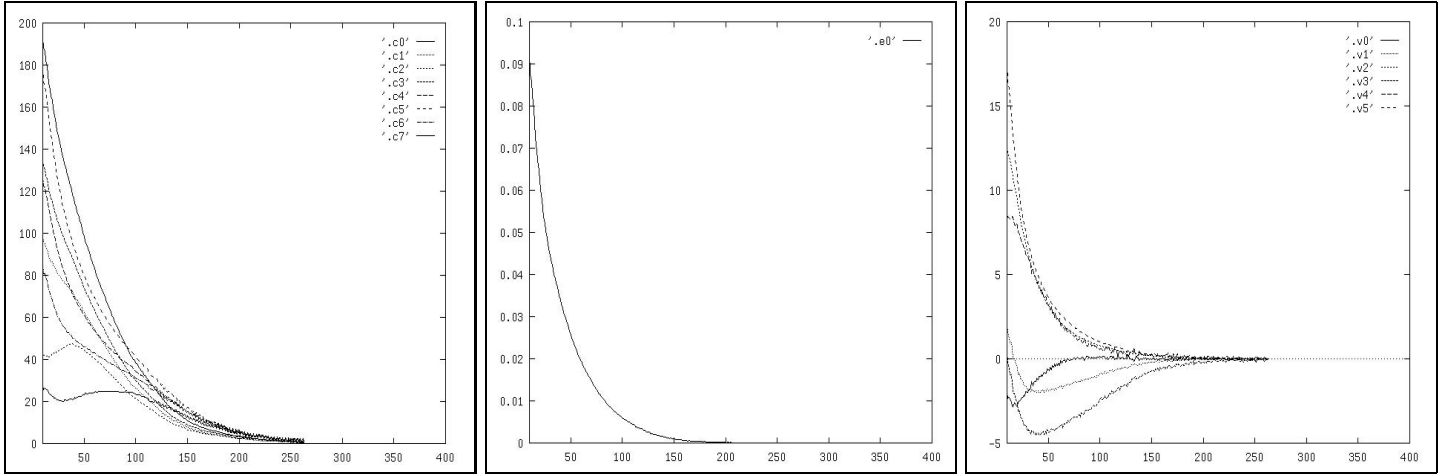
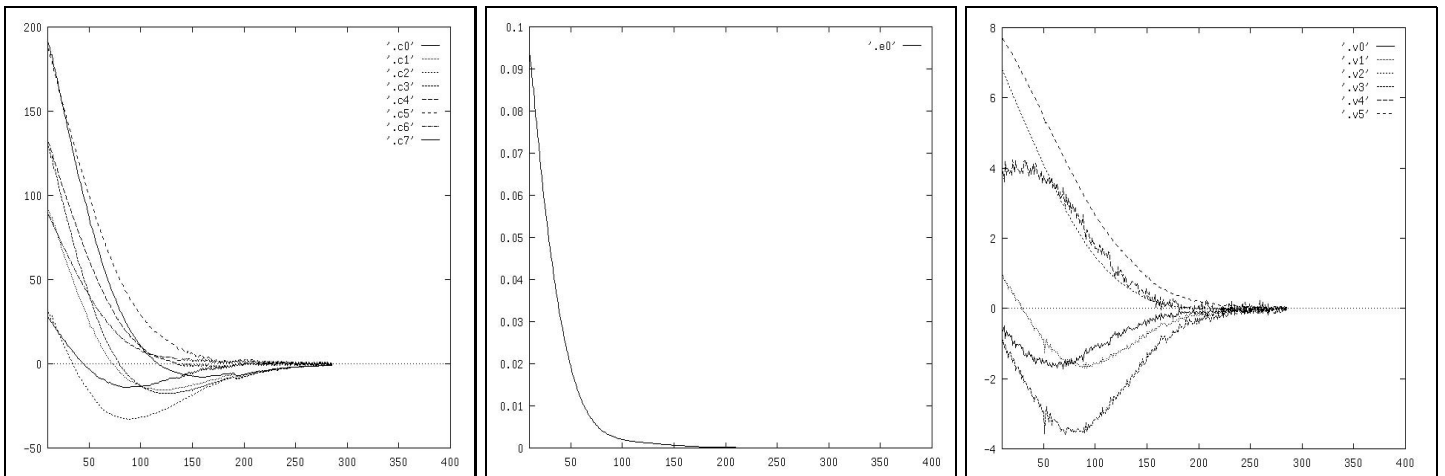
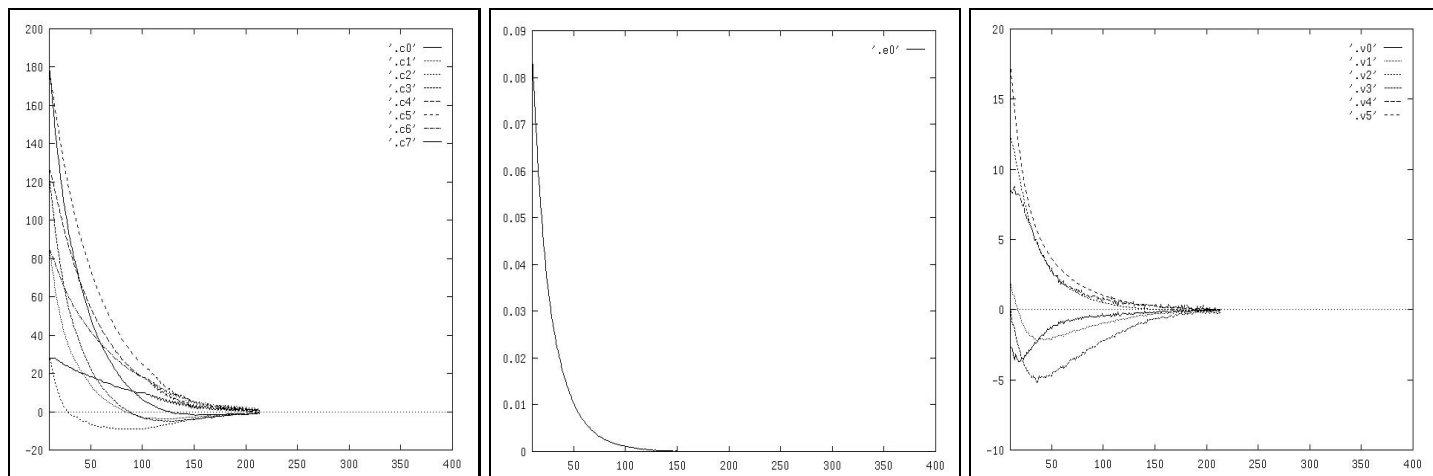
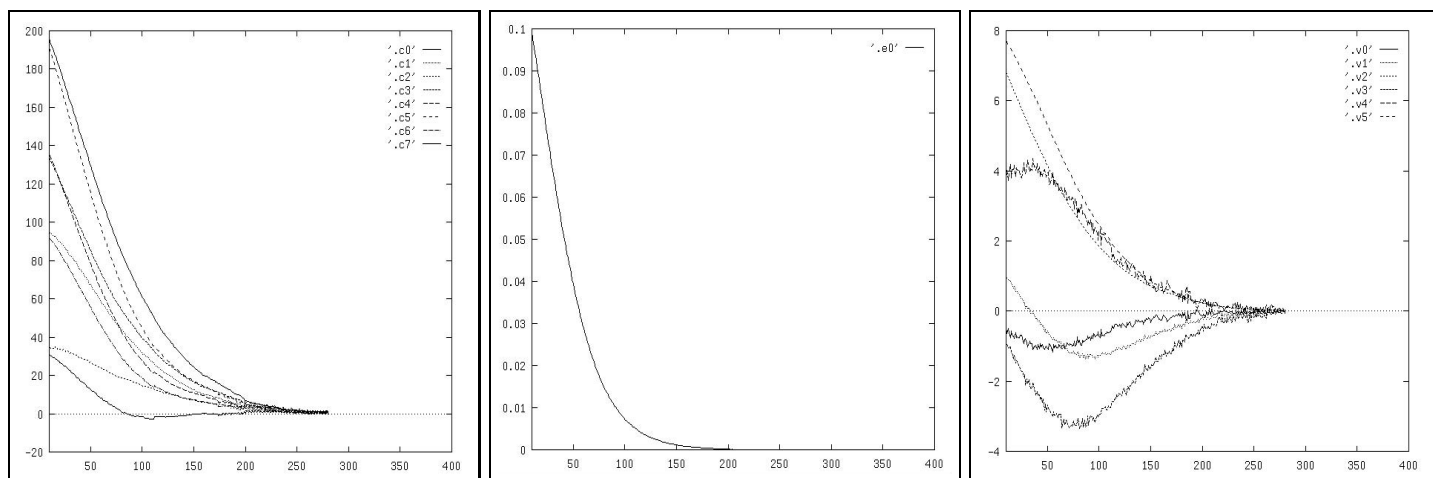
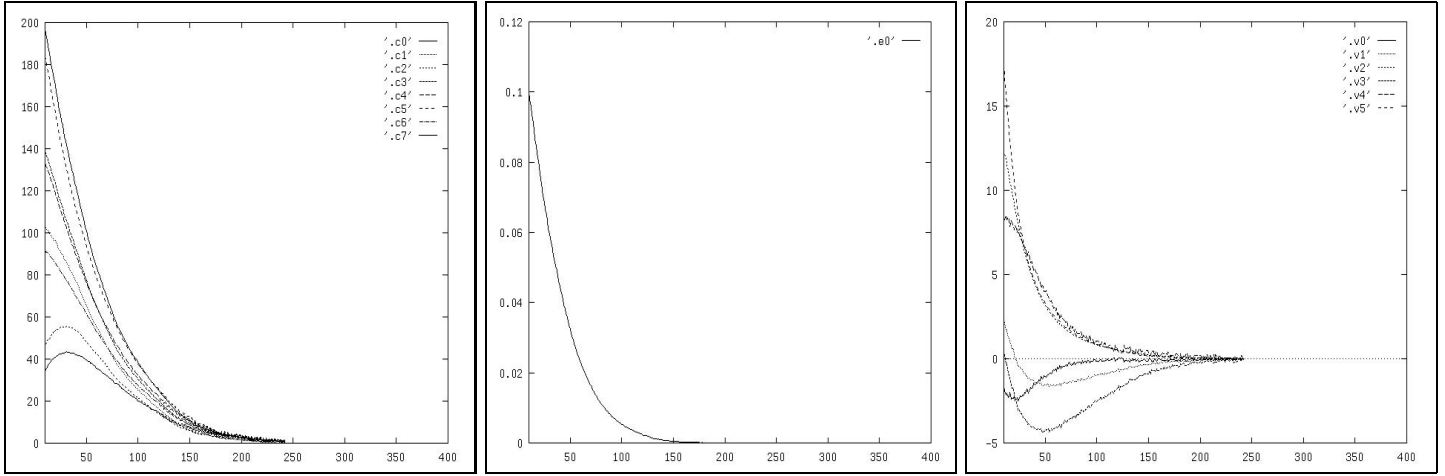
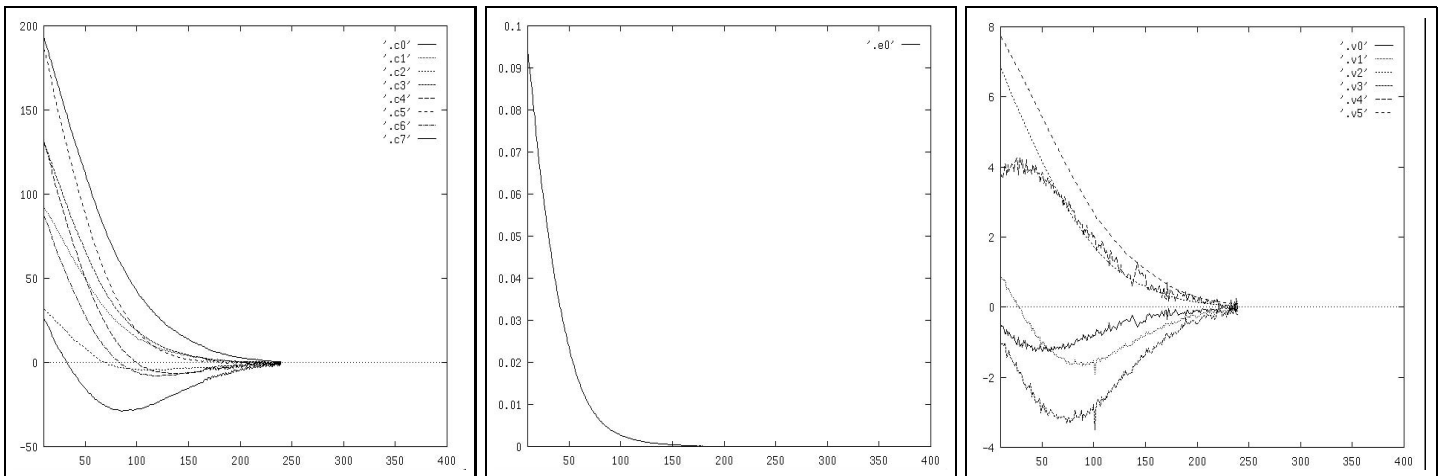


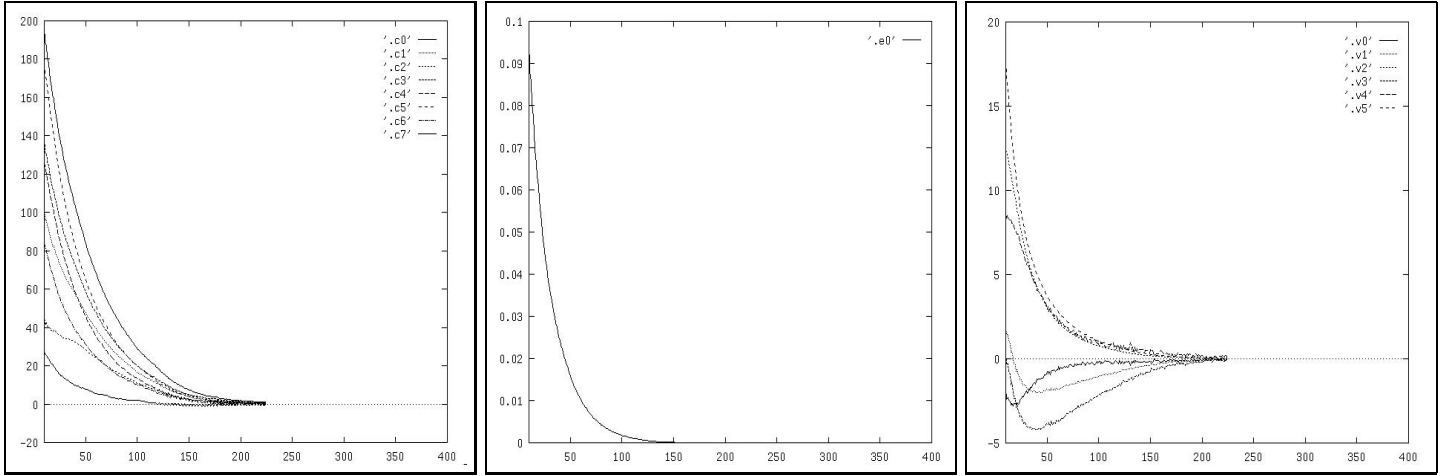
FIG. 9 - : Matrice non constante, erreur sur la focale (+50%)

FIG. 10 - : Matrice constante, erreur sur la translation en x

FIG. 11 - : Matrice non constante, erreur sur la translation en x FIG. 12 - : Matrice constante, erreur sur la translation en z

FIG. 13 - : Matrice non constante, erreur sur la translation en z FIG. 14 - : Matrice constante, erreur de rotation autour de x

FIG. 15 - : Matrice non constante, erreur de rotation autour de x FIG. 16 - : Matrice constante, erreur de rotation autour de z

FIG. 17 - : Matrice non constante, erreur de rotation autour de z

C.2.4 Erreurs combinées

Paramètres intrinsèques seuls Les valeurs suivantes des paramètres ont été utilisées à la place des valeurs exactes:

$$u_0 = 460, v_0 = 187, -fk_u = 920, -fk_v = 1380$$

Les résultats sont donnés figures 18 et 19.

Erreurs sur l'ensemble des paramètres On introduit des erreurs supplémentaires, à savoir sur la matrice de passage poignet du robot - caméra (de 5 dg sur chaque axe de rotation et de 5 cm sur chaque axe de translation). On obtient alors les résultats donnés figures 20 et 21.

On peut observer que la convergence est obtenue moins rapidement que dans les cas non faussés, mais plus rapidement que dans le cas précédent où l'on ne bruitait que les paramètres intrinsèques (les erreurs se compensent probablement dans ce cas).

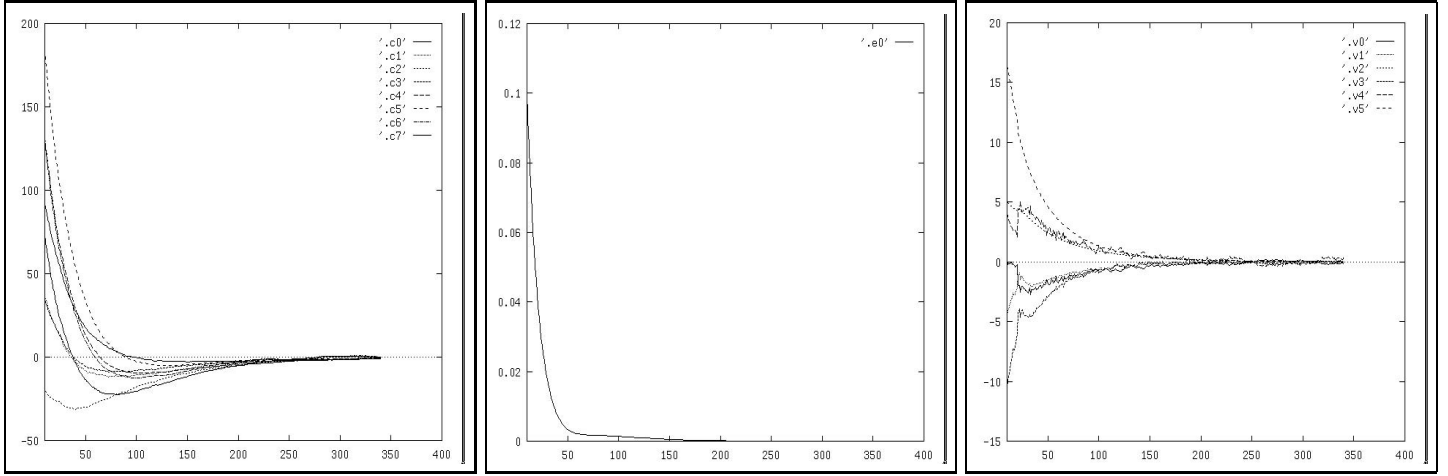
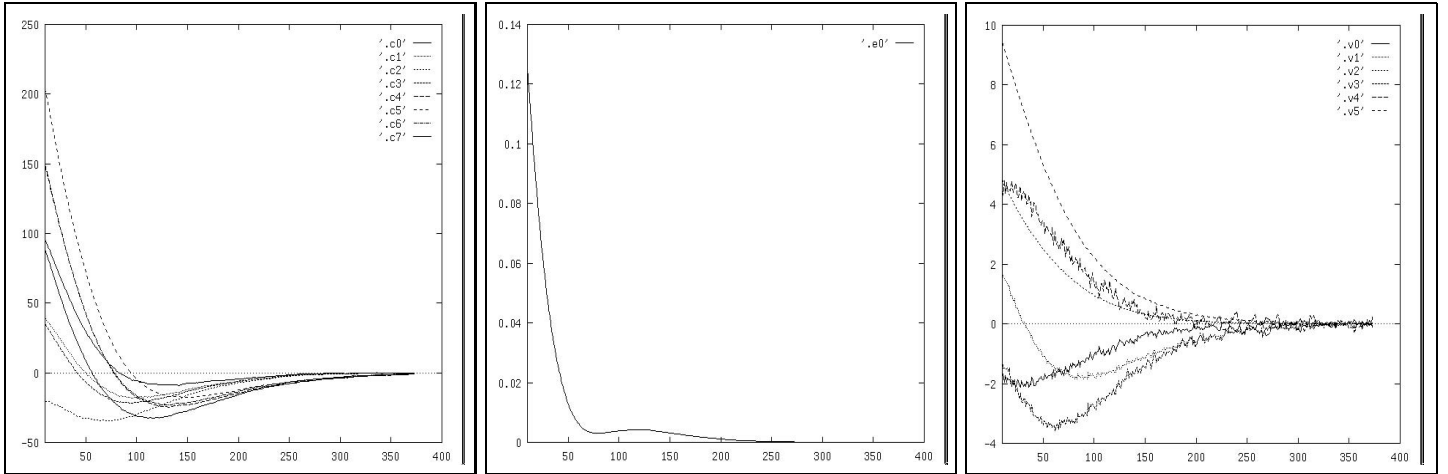


FIG. 18 - : Matrice non constante; erreurs sur les paramètres intrinsèques

FIG. 19 - : Matrice L^{-T} constante ($= L_{|s=s^*}^{-T}$)
Erreurs introduites sur les paramètres intrinsèques

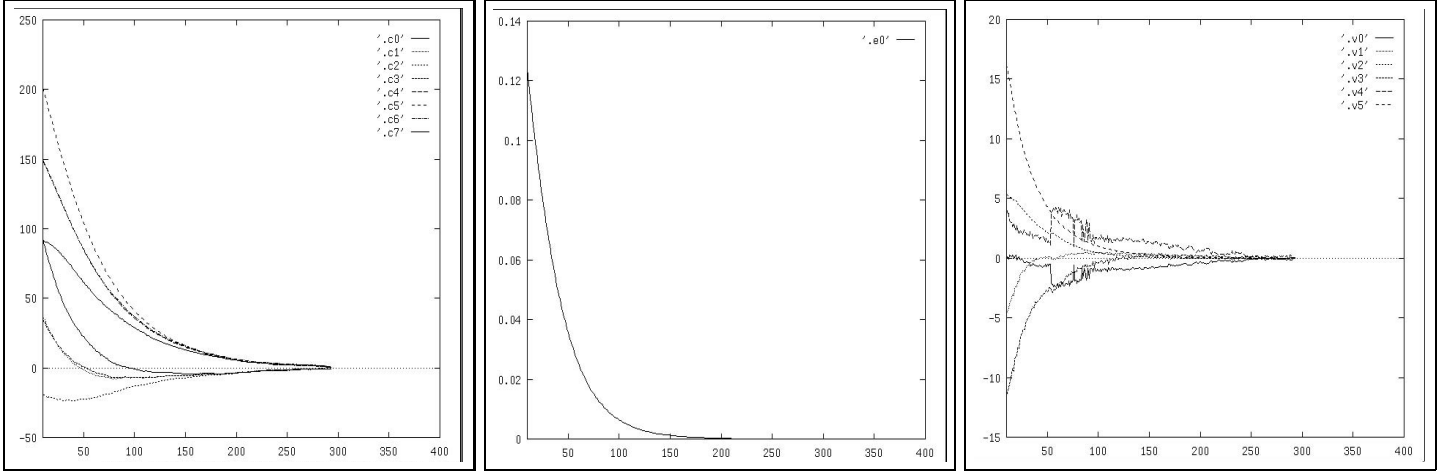


FIG. 20 - : Calcul de L^{-T} à chaque itération
Erreurs introduites sur les paramètres intrinsèques et la matrice de passage poignet-caméra

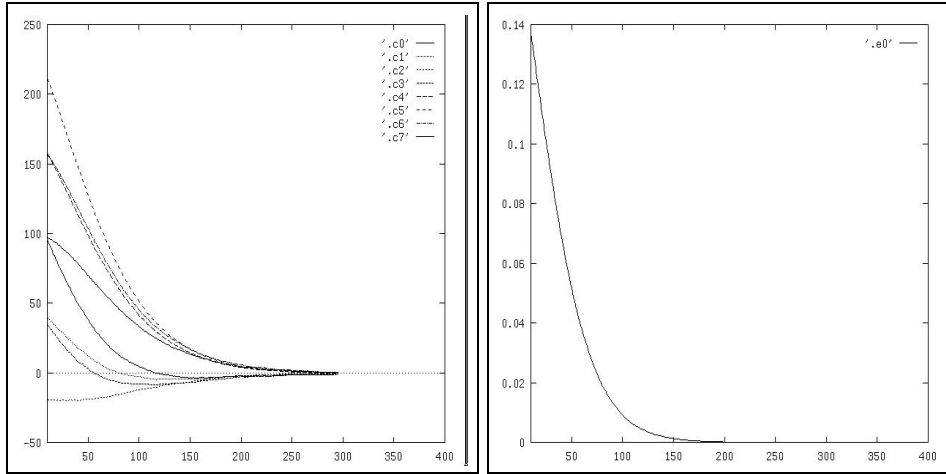


FIG. 21 - : Matrice L^{-T} constante ($= L_{|s=s*}^{-T}$)
Erreurs introduites sur les paramètres intrinsèques et la matrice de passage poignet-caméra

D Résultats de simulation

Dans les résultats présentés, on utilisera les notations suivantes:

- T : erreur initiale en translation seulement valant: $t_x = 10, t_y = 10, t_z = -20$
- TR : erreur initiale en translation et rotation valant: $t_x = 10, t_y = 20, t_z = -20$, angles d'Euler = $\{-5 \text{ deg}, 5 \text{ deg}, -10 \text{ deg}\}$
- V : la matrice \hat{L} utilisée dans le calcul de la commande est *variable*, c'est-à-dire calculée à chaque instant, en fonction des vrais paramètres 3D et avec les paramètres de calibration erronés.
- C : la matrice \hat{L} utilisée dans le calcul de la commande est *constante* et calculée en fonction des vrais paramètres 3D à l'équilibre et avec les paramètres de calibration erronés.

Les courbes présentées sont exclusivement la norme de l'erreur $\|s(t) - s^*\|$ et les 6 mouvements de la caméra en translation et rotation en fonction du temps. Comme s^* est calculé en fonction des paramètres de calibration corrects, $\|s(t) - s^*\|$ peut tendre vers zero même lorsque les paramètres de calibration utilisés dans la commande sont faux.

D.1 Série 1: références

Cette première série de résultats concerne le cas idéal, c'est-à-dire sans erreurs de calibration.

- *figure 1.1*: T ; V ; norme de l'erreur
- *figure 1.2*: T ; V ; mouvement
- *figure 1.3*: T ; C ; norme de l'erreur
- *figure 1.4*: T ; C ; mouvement
- *figure 1.5*: TR ; V ; norme de l'erreur
- *figure 1.6*: TR ; V ; mouvement
- *figure 1.7*: TR ; C ; norme de l'erreur
- *figure 1.8*: TR ; C ; mouvement

On notera une convergence plus lente dans les cas C .

D.2 Erreurs sur les paramètres intrinsèques

D.2.1 Série 2: effet d'une erreur sur la distance focale seule

On note $a = \frac{f}{f}$

- *figure 2.1*: $a = \{0.5 ; 1.2 ; 2 ; 3 ; \}$; T ; V ; norme de l'erreur
- *figure 2.2*: $a = \{0.5 ; 1.2 ; 2 ; 3 ; \}$; T ; C ; norme de l'erreur
- *figure 2.3*: $a = 3$; TR ; C ; norme de l'erreur
- *figure 2.4*: $a = 3$; TR ; C ; mouvement

Dans ces résultats l'effet de a apparaît essentiellement dans la vitesse de convergence.

D.2.2 Série 3: effet d'une erreur sur l'angle des axes image seule

- *figure 3.1*: T ; C et V ; $\theta = \{60 \text{ deg} ; 120 \text{ deg} \}$; norme de l'erreur
- *figure 3.2*: TR ; C ; $\theta = \{45 \text{ deg} ; 135 \text{ deg} \}$; norme de l'erreur

On notera le peu d'influence de ce paramètre.

D.2.3 Série 4: effet d'une erreur sur l'origine des axes image seule

- *figure 4.1*: TR ; C et V ; $\text{offset} = \{0 ; 0.2\}$; norme de l'erreur

On notera le peu d'influence de ce paramètre.

D.3 Erreurs sur les paramètres extrinsèques

D.3.1 Série 5: effet d'une erreur sur la translation seule

- T ;

Les erreurs en x et y n'ayant pas d'influence, on fait seulement varier la translation en z : $t_3 = \{-20 ; 0 \text{ } 20\}$.

- *figure 5.1*: V ; norme de l'erreur
- *figure 5.2*: C ; norme de l'erreur

- TR ;

On a:

$$\begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix} = \left\{ \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} ; \begin{pmatrix} -20 \\ -20 \\ 0 \end{pmatrix} ; \begin{pmatrix} 0 \\ 0 \\ -20 \end{pmatrix} \right\}$$

- *figure 5.3*: V ; norme de l'erreur

- *figure 5.4*: C; norme de l'erreur

Bien que les erreurs en x et y aient dans ce cas une certaine influence, celle de l'erreur en z est nettement prépondérante.

D.3.2 Série 6: effet d'une erreur sur l'orientation seule

On représente l'erreur d'orientation par l'angle ϕ et le vecteur unitaire u de la rotation associée.

- $\phi = 30 \text{ deg}; u = \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} ; \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} ; \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}$

- *figure 6.1*: T; V; norme de l'erreur
- *figure 6.2*: T; C; norme de l'erreur
- *figure 6.3*: TR; V; norme de l'erreur

- $u = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$

- *figure 6.4* $\phi = \{20; 25; 26; 30\}$; TR; C; norme de l'erreur
- *figure 6.5* $\phi = \{26; 30; 45\}$; TR; V; norme de l'erreur

- $u = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$

- *figure 6.6* $\phi = \{-60; -45; -20; 20; 45; 60\}$; TR; V; norme de l'erreur
- *figure 6.7* $\phi = \{20; 45; 60\}$; TR; C; norme de l'erreur

La figure 6.4 montre un comportement étrange de l'évolution de l'erreur à partir de 25 deg: après un début de convergence correct, l'erreur augmente très rapidement puis (sauf pour $\phi > 30 \text{ deg}$ où il y a divergence) retrouve une décroissance exponentielle. Ce comportement, pour lequel nous n'avons pu trouver d'explication précise, ne se produit pas avec la commande à matrice variable (figure 6.5), bien que dans ce dernier cas une erreur importante affecte fortement la convergence.

D.4 Erreurs combinées

D.4.1 Série 7: effet de petites erreurs combinées, hors offset

Les paramètres utilisés sont:

- $a=1.5$

- $\theta = 80$ deg
- $\phi = 10$ deg
- $u^T = (\frac{1}{\sqrt{3}} \frac{1}{\sqrt{3}} \frac{1}{\sqrt{3}})$
- $t^T = (10 \ 10 \ 10)$

Les résultats présentés sont:

- *figure 7.1*: TR ; V ; norme de l'erreur; la courbe de référence est rappelée.
- *figure 7.2*: TR ; C ; norme de l'erreur
- *figure 7.3*: TR ; C ; mouvement

D.4.2 Série 8: effet de grandes erreurs combinées, hors offset

Les paramètres communs à l'ensemble des essais de cette section sont:

- $\theta = 60$ deg
- $\phi = 30$ deg
- $t^T = (20 \ -20 \ -20)$

Première partie On a:

- $a = 2.5$
- $u^T = (\frac{1}{\sqrt{3}} \frac{1}{\sqrt{3}} \frac{1}{\sqrt{3}})$

Les résultats sont les suivants:

- *figure 8.1*: V ; norme de l'erreur. La position/orientation initiale est donnée dans le tableau ci-dessous pour chacune des courbes tracées:

Courbe	T	R
1	10 10 -20	0 0 0
2	10 20 -20	-5 5 -10
3	0 0 0	30 -30 20
4	0 0 0	30 30 30

- *figure 8.2*: C ; norme de l'erreur. La position/orientation initiale est donnée dans le tableau ci-dessous pour chacune des courbes tracées:

Courbe	T	R
1	10 10 - 20	0 0 0
2	10 20 - 20	-5 5 - 10
3	0 0 0	30 - 30 20
4	0 0 0	40 40 40
5	30 30 - 40	0 0 0
6	0 0 0	30 30 30

Les cas nos 4 et 5 de la figure 8.2 n'apparaissent pas dans la figure 8.1 car ils conduisent alors à une divergence. Ici, le choix "matrice de commande constante" apparaît meilleur que le choix "variable".

Deuxième partie On a :

- $a=0.5$
- $u^T = (0 \ 0 \ 1)$

Les résultats sont les suivants:

- *figure 8.3*: V ; norme de l'erreur. La position/orientation initiale est donnée dans le ta-

bleau ci-dessous pour chacune des courbes tracées:

Courbe	T	R
1	30 30 - 40	0 0 0
2	0 0 0	20 20 20
3	30 30 - 40	20 20 20
4	30 30 - 40	25 25 25

- *figure 8.4*: C ; norme de l'erreur. La position/orientation initiale est donnée dans le tableau ci-dessous pour chacune des courbes tracées:

Courbe	T	R
1	0 0 0	30 30 30
2	30 30 - 40	0 0 0
3	30 30 - 40	20 20 20
4	40 40 - 20	25 25 25

Les cas 1 et 4 de la figure 8.4 conduisent à une divergence avec le choix "matrice variable". On a donc comme précédemment un meilleur comportement avec une matrice de commande constante.

D.4.3 Série 9: effet d'erreurs combinées réalistes

Les paramètres utilisés sont:

- $a=0.7$
- $\theta = 89 \text{ deg}$

- $\phi = 10$ deg
- $u^T = (0 \ 0 \ 1)$
- $t^T = (5 \ -5 \ 10)$

Pas d'erreur d'offset La position/orientation initiale est donnée dans le tableau ci-dessous pour chacune des courbes tracées:

Courbe	T	R
1	0 0 0	10 10 10
2	10 10 - 10	0 0 0
3	10 10 - 10	10 10 10

Les résultats présentés sont:

- *figure 9.1*: rappel de la référence: V ; norme de l'erreur
- *figure 9.2*: V ; norme de l'erreur
- *figure 9.3*: C ; norme de l'erreur

Présence d'une erreur d'offset

- *figure 9.4*: offset = 0.1 ; V ; les conditions initiales sont celles du cas 3 ci-dessus; norme de l'erreur

On constate dans tous les cas un bon comportement de l'erreur: la convergence est peu modifiée en transitoire et en durée.

D.5 Série 10: effet de grandes erreurs avec des conditions initiales proches de l'équilibre

D.5.1 matrice de commande variable (V)

1- Ecart initial $TR = \{ 0 \ 0 \ 0 \ ; \ 5 \ 5 \ 5 \}$

- *figure 10.1*: norme de l'erreur. Les divers cas de choix paramétriques effectués sont indiqués dans le tableau ci-dessous:

Courbe	α	θ	ϕ	u	t	offset
1 (réf.)	1	90	0		0	0
2	3	80	60	0 0 1	10 10 -10	0
3	0.4	80	45	0 0 1	10 10 -10	0

- *figure 10.2*: norme de l'erreur. Même paramètres que la courbe 2 précédente, mais avec offset de 0.2.

2- Ecart initial $TR = \{ 5 \ 5 \ -5; 0 \ 0 \ 0 \}$

- *figure 10.3*: norme de l'erreur. Les divers cas de choix paramétriques effectués sont indiqués dans le tableau ci-dessous:

Courbe	α	θ	ϕ	u	t	offset
1 (réf.)	1	90	0		0	0
2	3	80	60	0 0 1	10 10 -10	0
3	0.4	80	45	0 0 1	10 10 -10	0
4	2	80	45	0 0 1	10 10 -10	0

3- Ecart initial $TR = \{ 5 \ 5 \ -5; 5 \ 5 \ 5 \}$

- *figure 10.4*: norme de l'erreur. Les divers cas de choix paramétriques effectués sont indiqués dans le tableau ci-dessous:

Courbe	α	θ	ϕ	u	t	offset
1	3	80	60	0 0 1	10 10 -10	0
2	0.4	80	45	0 0 1	10 10 -10	0

- *figure 10.5*: mouvements pour le cas 2 de la figure 10.4.

D.5.2 matrice de commande constante (C)**4- Ecart initial $TR = \{ 0 \ 0 \ 0 ; 5 \ 5 \ 5 \}$**

- *figure 10.6*: norme de l'erreur. Les divers cas de choix paramétriques effectués sont indiqués dans le tableau ci-dessous:

Courbe	α	θ	ϕ	u	t	offset
1	3	80	60	0 0 1	10 10 -10	0
2	0.4	80	45	0 0 1	10 10 -10	0

5- Ecart initial $TR = \{ 5 \ 5 \ -5; 0 \ 0 \ 0 \}$

- *figure 10.7*: norme de l'erreur. Les divers cas de choix paramétriques effectués sont indiqués dans le tableau ci-dessous:

Courbe	α	θ	ϕ	u	t	offset
1 (réf.)	1	90	0		0	0
2	0.4	80	45	0 0 1	10 10 -10	0
3	2	80	45	0 0 1	10 10 -10	0

- *figure 10.8*: mouvements pour le cas 2 de la figure 10.4, mais avec matrice constante.

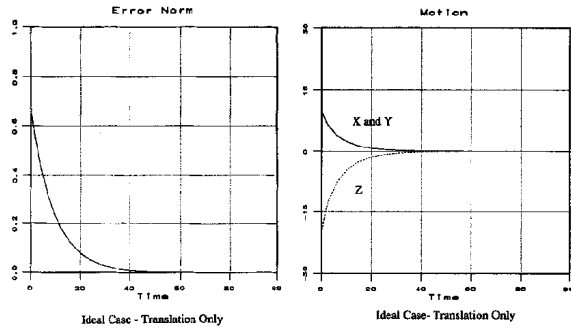


figure 1.1

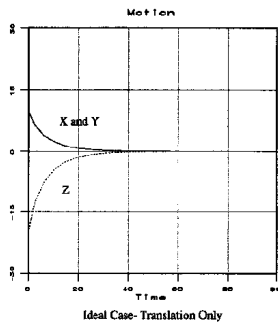


figure 1.2

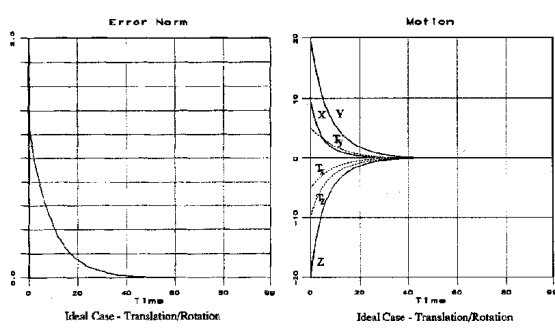


figure 1.5

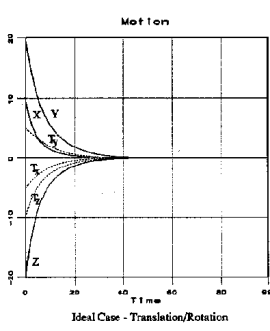


figure 1.6

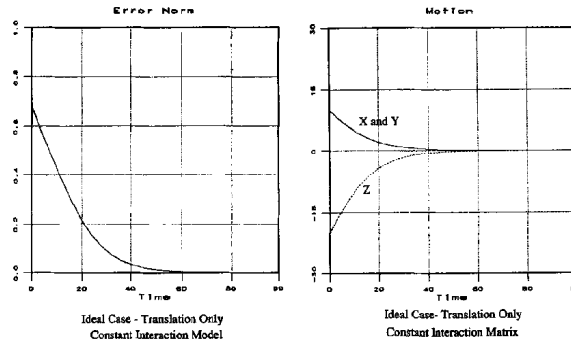


figure 1.3

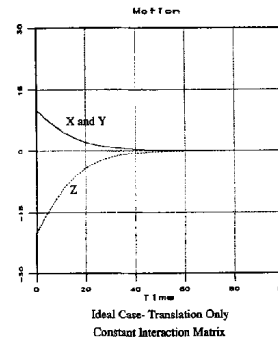


figure 1.4

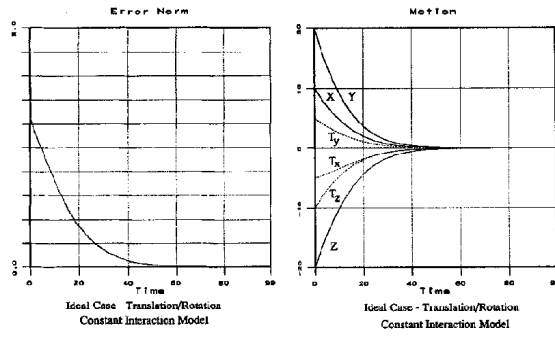


figure 1.7

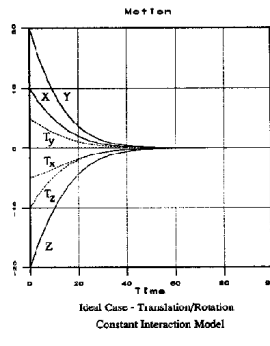


figure 1.8

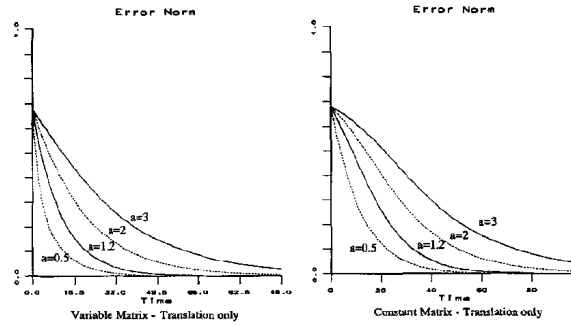


figure 2.1

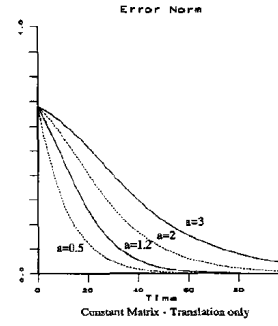


figure 2.2

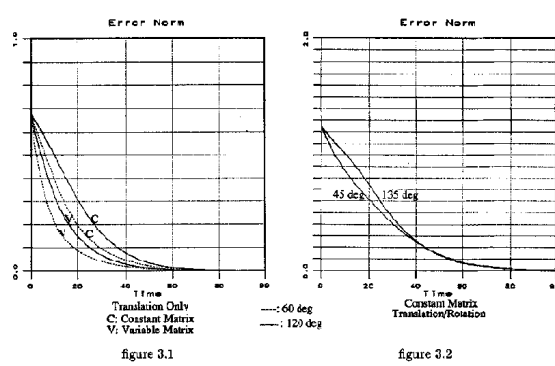


figure 3.1

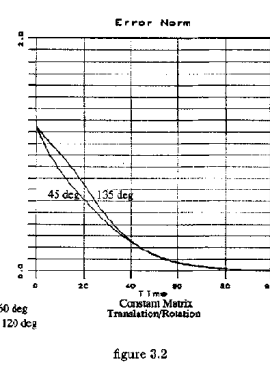


figure 3.2

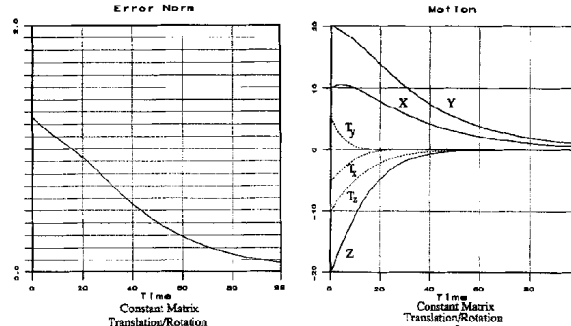


figure 2.3

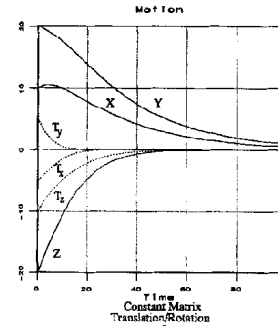


figure 2.4

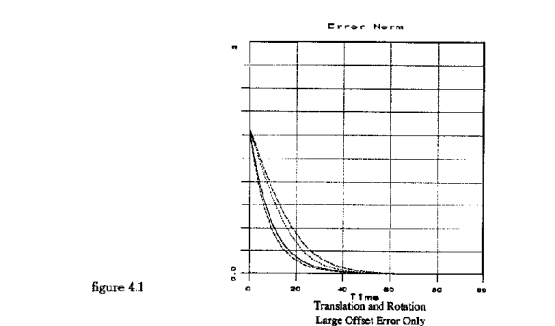


figure 4.1

Offset=0 Offset=0.2 Offset=0.4
Variable Matrix Constant Matrix

INRIA

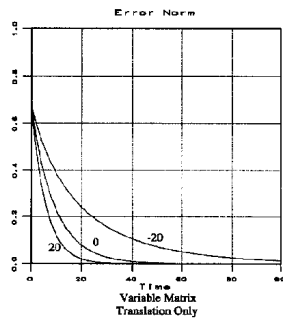


figure 5.1

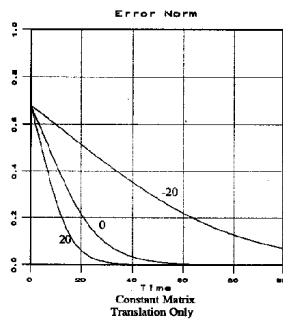


figure 5.2

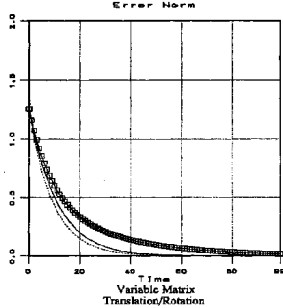


figure 5.3

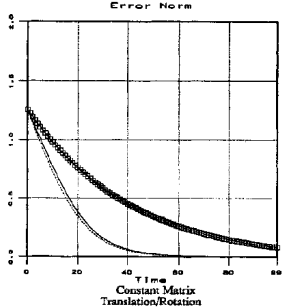


figure 5.4

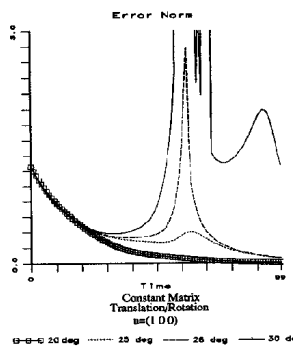


figure 6.4

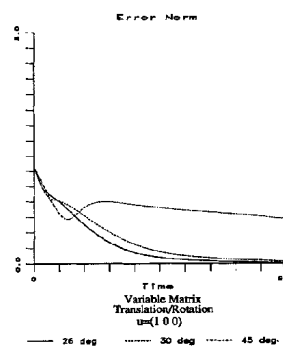


figure 6.5

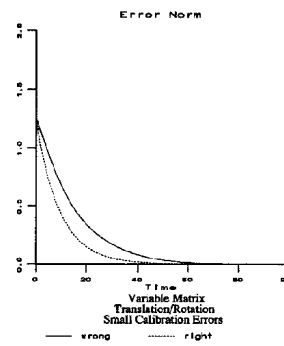


figure 7.1

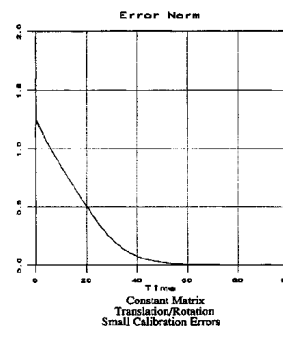


figure 7.2

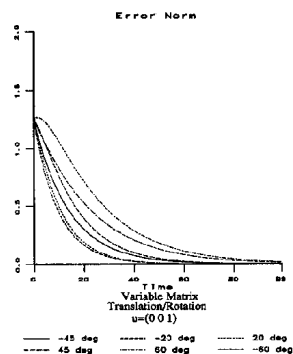


figure 6.6

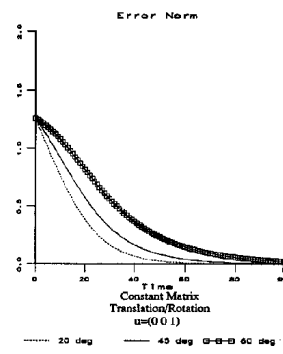


figure 6.7



figure 7.3

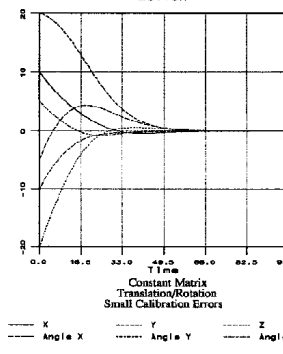


figure 7.4

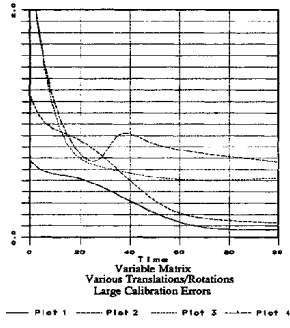


figure 8.1

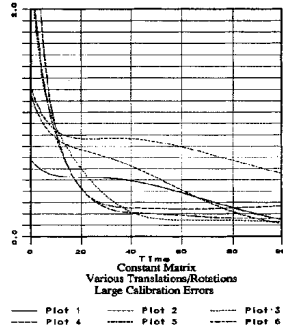


figure 8.2

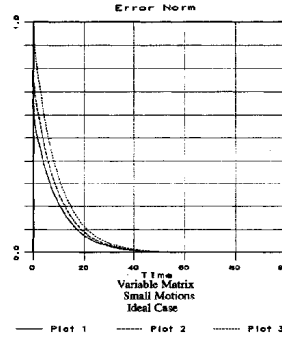


figure 9.1

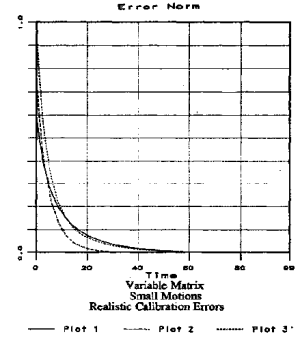


figure 9.2

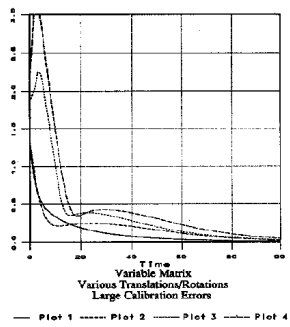


figure 8.3

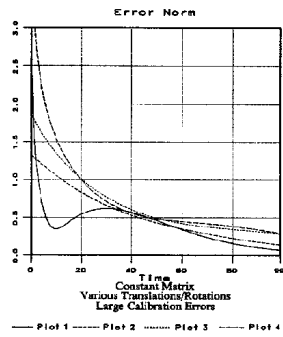


figure 8.4

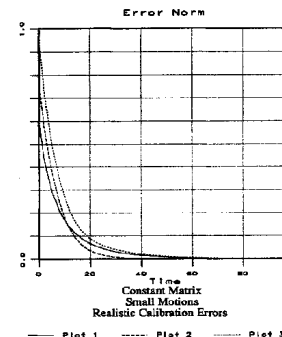


figure 9.3

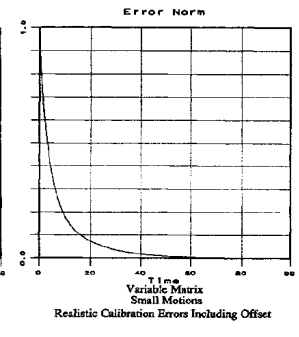


figure 9.4

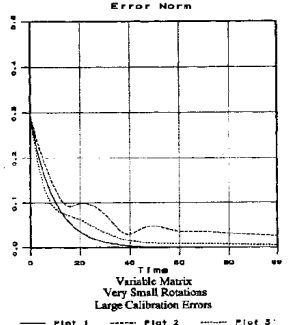


figure 10.1

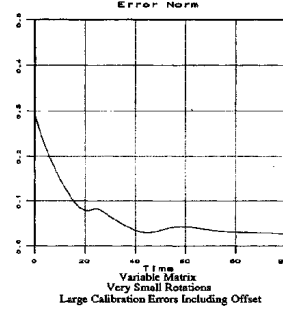


figure 10.2

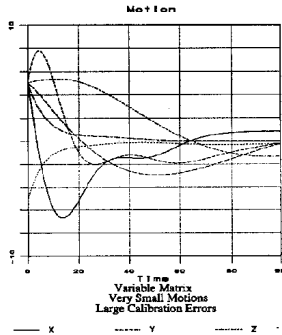


figure 10.5

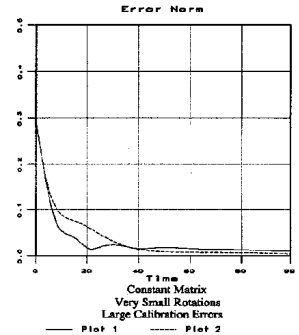


figure 10.6

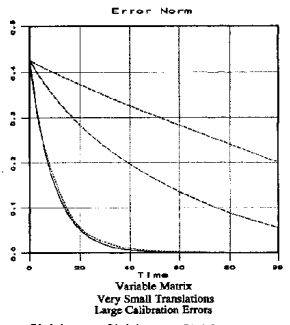


figure 10.3

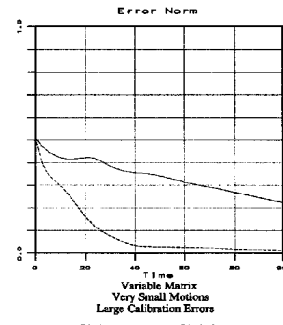


figure 10.4

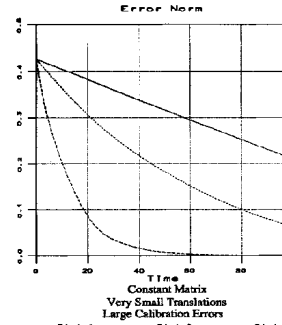


figure 10.7

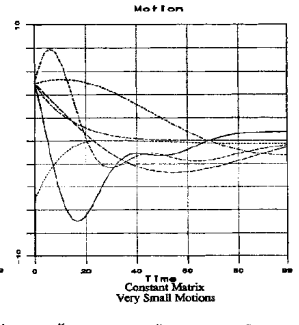


figure 10.8

Table des matières

1	Introduction	3
2	L'approche "asservissement visuel"	3
2.1	Modélisation	4
2.2	Une approche du contrôle	5
2.2.1	Préliminaires	5
2.2.2	Forme générale	5
2.3	Problème considéré	7
3	Formes de la matrice d'interaction	7
3.1	Calcul direct	7
3.2	Forme "canonique"	8
4	Influence des erreurs	10
4.1	Erreurs de calibration	10
4.1.1	Embryon d'étude analytique	10
4.1.2	Etude en simulation	13
4.1.3	Etude expérimentale	15
4.2	Erreurs de positionnement final dues aux incertitudes de calibration	15
4.2.1	Position du problème	15
4.2.2	En pratique	21
4.3	Effet des incertitudes sur le 3D	22
4.3.1	Calibration connue	22
4.3.2	Calibration inconnue	25
5	Conclusion	25
A	Asservissement visuel en stéréovision	30
B	Un invariant différentiel	33
C	Résultats expérimentaux	36
C.1	Conditions Expérimentales	36
C.2	Résultats	37
C.2.1	Courbes de référence	37
C.2.2	Erreurs sur la distance focale	37
C.2.3	Erreurs sur les paramètres extrinsèques	37
C.2.4	Erreurs combinées	44

D Résultats de simulation	47
D.1 Série 1: références	47
D.2 Erreurs sur les paramètres intrinsèques	48
D.2.1 Série 2: effet d'une erreur sur la distance focale seule	48
D.2.2 Série 3: effet d'une erreur sur l'angle des axes image seule	48
D.2.3 Série 4: effet d'une erreur sur l'origine des axes image seule	48
D.3 Erreurs sur les paramètres extrinsèques	48
D.3.1 Série 5: effet d'une erreur sur la translation seule	48
D.3.2 Série 6: effet d'une erreur sur l'orientation seule	49
D.4 Erreurs combinées	49
D.4.1 Série 7: effet de petites erreurs combinées, hors offset	49
D.4.2 Série 8: effet de grandes erreurs combinées, hors offset	50
D.4.3 Série 9: effet d'erreurs combinées réalistes	51
D.5 Série 10: effet de grandes erreurs avec des conditions initiales proches de l'équilibre	52
D.5.1 matrice de commande variable (V)	52
D.5.2 matrice de commande constante (C)	53



Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irista, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 46 avenue Félix Viallet, 38031 GRENOBLE Cedex 1
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399